Utilizing CDISC CORE Validation Tools for Clinical Submission Domains

Presented by Daniel Christen, Life Sciences Industry Expert at SAS Institute AG
May 15th 2025

Meet the Speaker

Daniel Christen

Title: Life Sciences Industry Expert

Organization: SAS Institute AG

Daniel Christen is a life sciences industry expert at SAS, a leading provider of analytics software and solutions for the life sciences sector. Since his start in 2010, he has played a pivotal role in empowering SAS clientele to navigate the extensive data landscape for enhanced business value. His areas of emphasis include the modernization of processes, fostering data transparency, and addressing pharmaceutical development challenges, all contributing to the conversion of complex data into groundbreaking insights.

Disclaimer and Disclosures

- *The views and opinions expressed in this presentation are those of the author(s) and do not necessarily reflect the official policy or position of CDISC.*

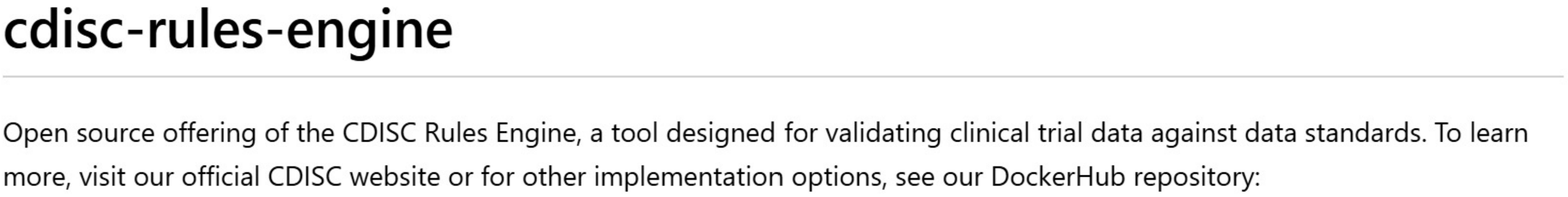
- *The author(s) have no real or apparent conflicts of interest to report.*

Agenda

1. Use Case Overview
2. Importing data to your repository
3. Connecting your repository to VS Code
4. Making your data available in VS Code
5. Connecting the CDISC CORE Rules Engine
6. Setting up the Python environment for the CDISC CORE Rules Engine
7. Executing the CDISC CORE Rules Engine
8. Writing the final report to your repository under version control
9. Q&A

Use Case Overview

CDISC CORE Rules Engine

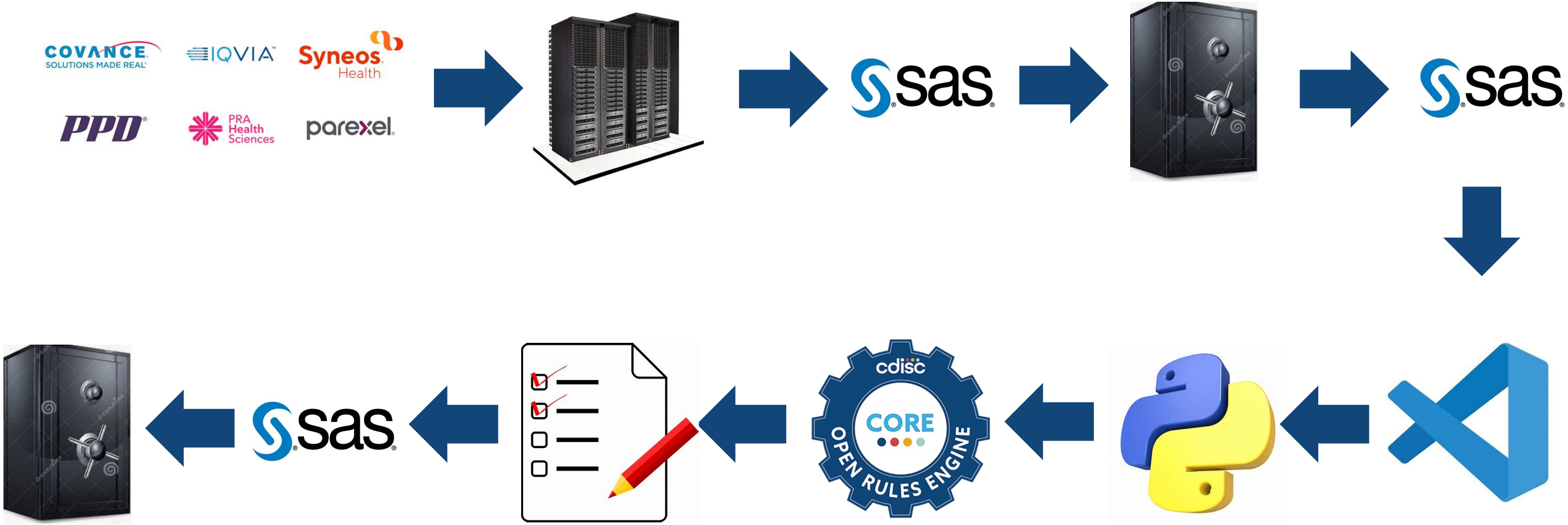


cdisc-rules-engine

Open source offering of the CDISC Rules Engine, a tool designed for validating clinical trial data against data standards. To learn more, visit our official CDISC website or for other implementation options, see our DockerHub repository:

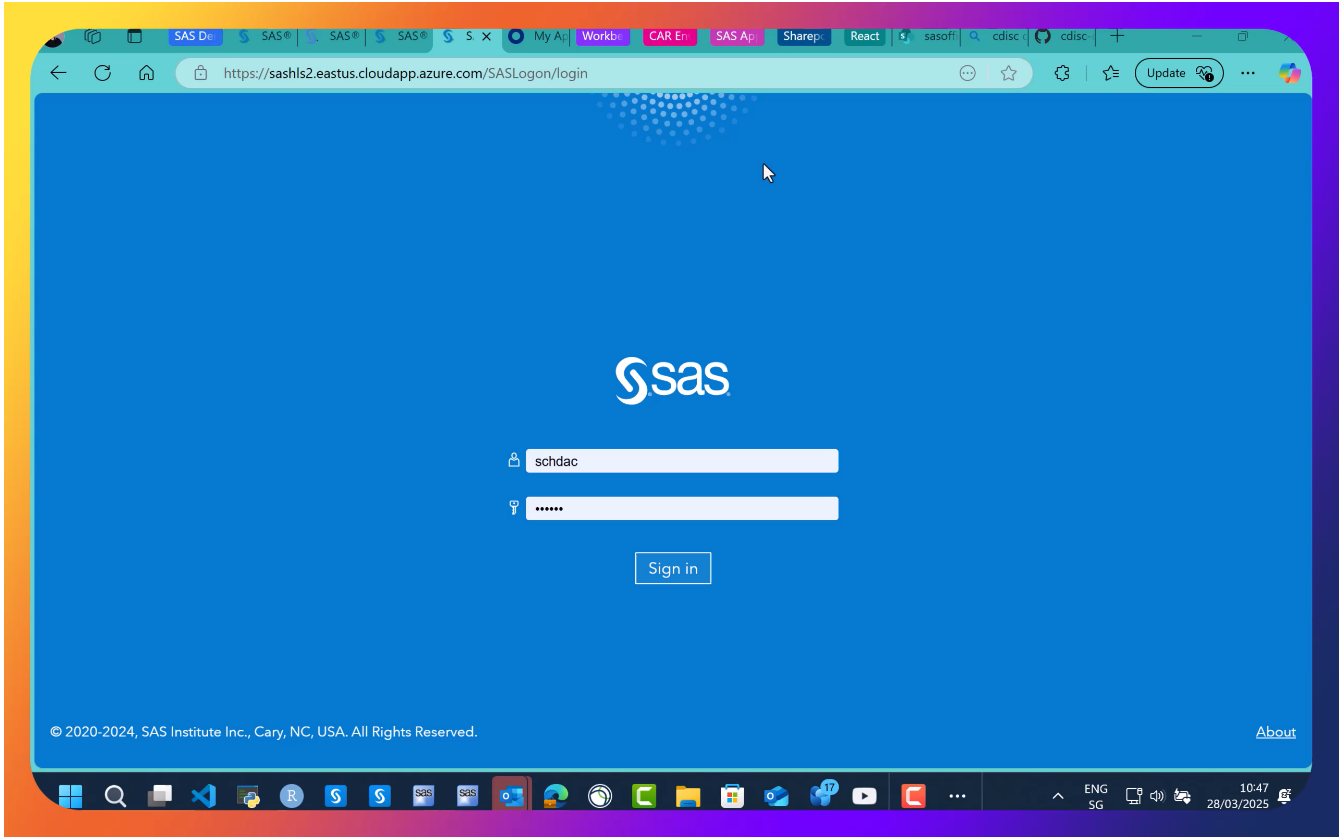
https://github.com/cdisc-org/cdisc-rules-engine

Use Case Overview



Importing data to your repository

Importing data into the repository



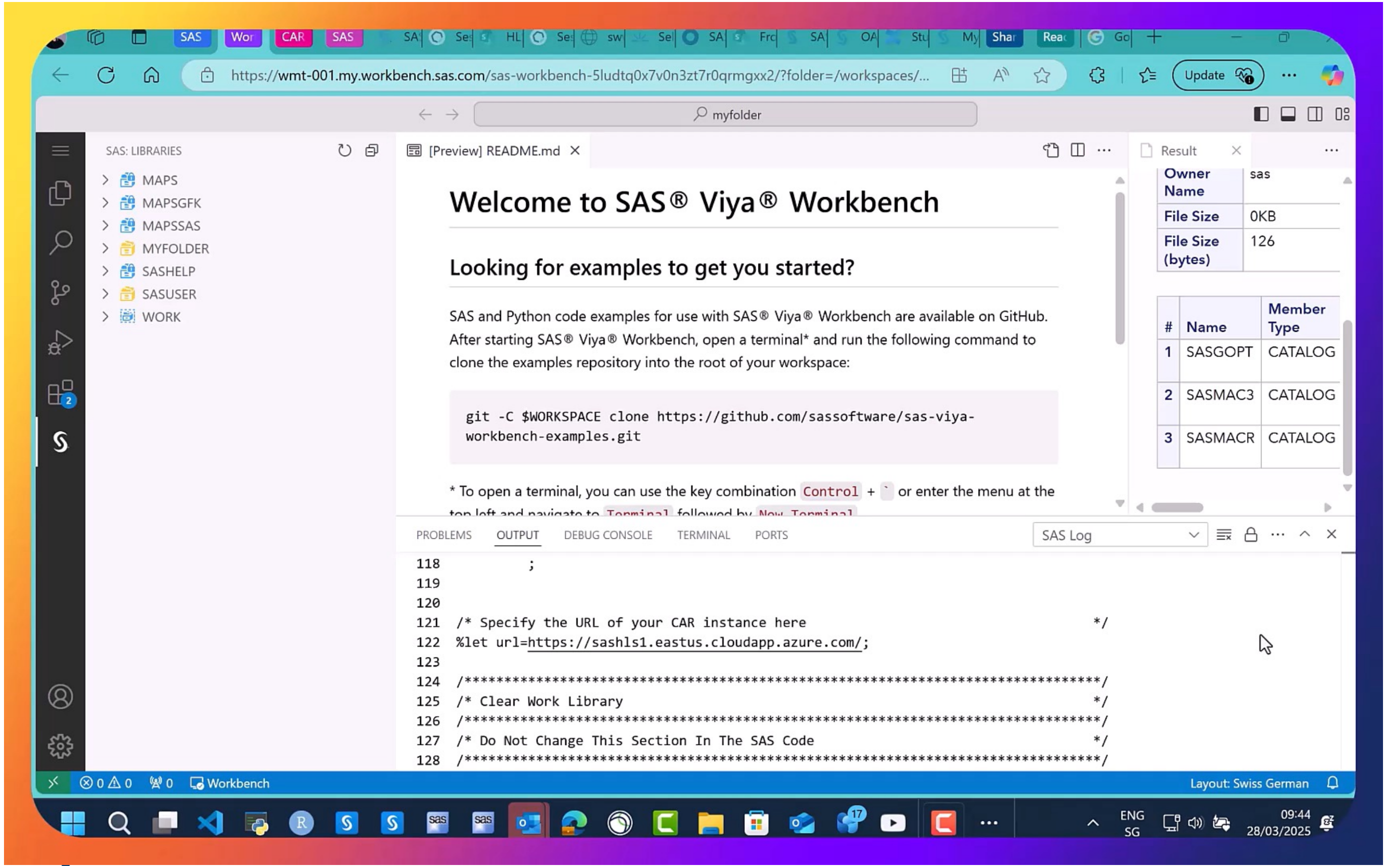
schdac

••••••

© 2020-2024, SAS Institute Inc., Cary, NC, USA. All Rights Reserved.

Connecting your repository to VS Code

Connecting your repository to VS Code



Welcome to SAS® Viya® Workbench

Looking for examples to get you started?

SAS and Python code examples for use with SAS® Viya® Workbench are available on GitHub. After starting SAS® Viya® Workbench, open a terminal* and run the following command to clone the examples repository into the root of your workspace:

```
git -C $WORKSPACE clone https://github.com/sassoftware/sas-viya-
workbench-examples.git
```

* To open a terminal, you can use the key combination `Control` + ` or enter the menu at the top left and navigate to `Terminal` followed by `New Terminal`

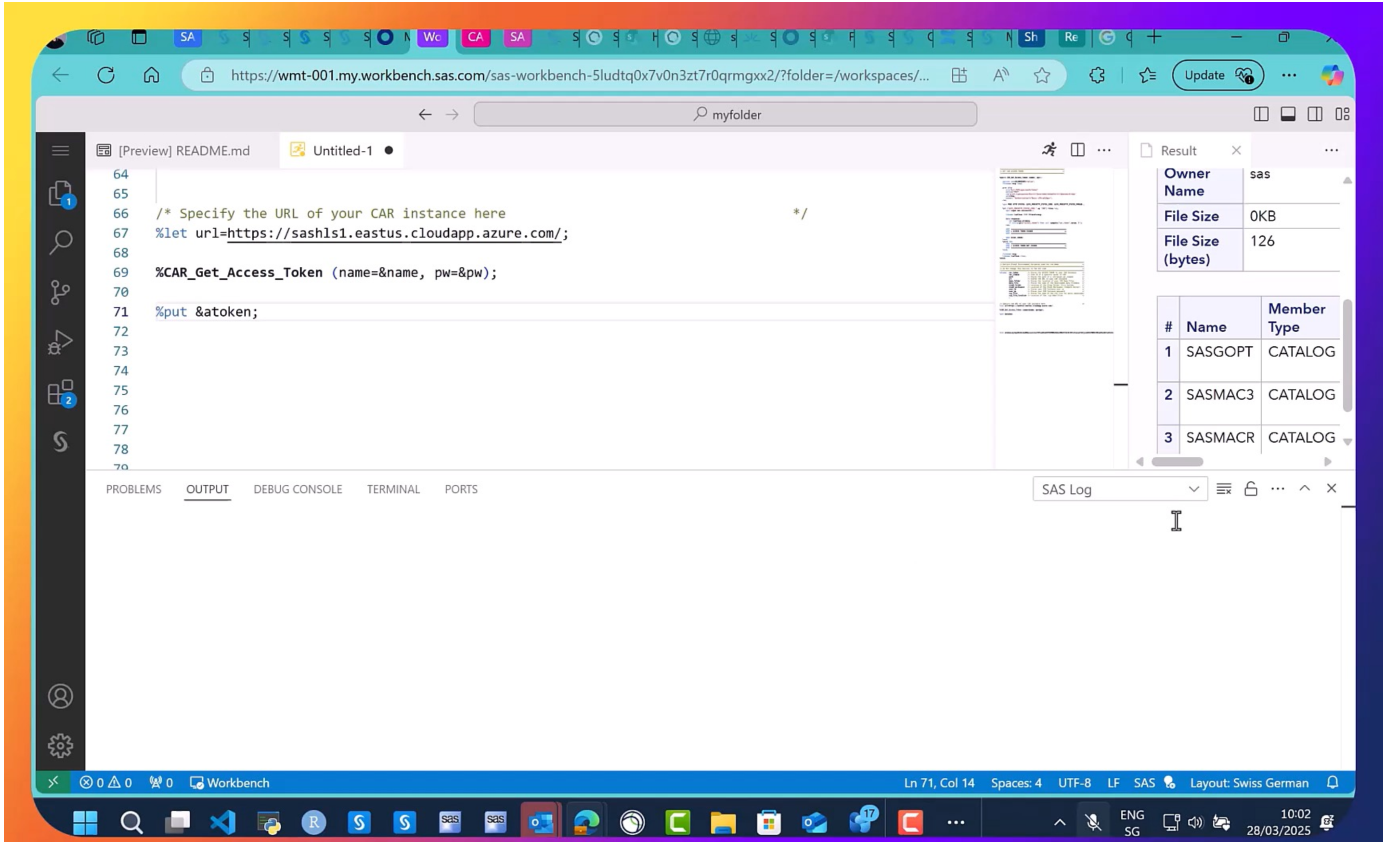
Owner Name	sas
File Size	0KB
File Size (bytes)	126

#	Name	Member Type
1	SASGOPT	CATALOG
2	SASMAC3	CATALOG
3	SASMACR	CATALOG

```
118         ;
119
120
121  /* Specify the URL of your CAR instance here                                   */
122  %let url=https://sashls1.eastus.cloudapp.azure.com/;
123
124  /****************************************************************************/
125  /* Clear Work Library                                                         */
126  /****************************************************************************/
127  /* Do Not Change This Section In The SAS Code                                 */
128  /****************************************************************************/
```

Making your data available in VS Code

Making your data available in VS Code



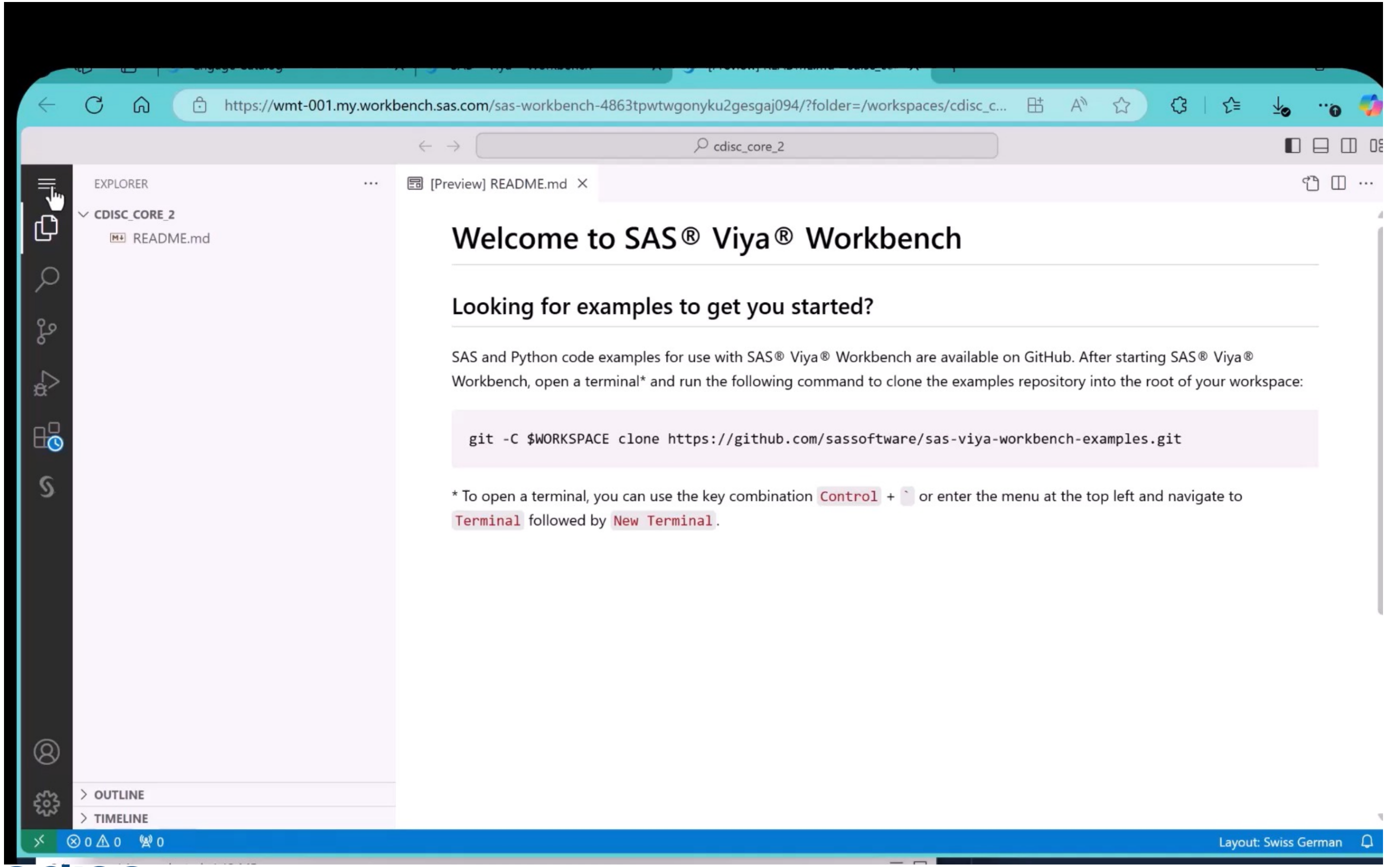
```
/* Specify the URL of your CAR instance here                                   */
%let url=https://sashls1.eastus.cloudapp.azure.com/;

%CAR_Get_Access_Token (name=&name, pw=&pw);

%put &atoken;
```

Owner Name	sas
File Size	0KB
File Size (bytes)	126

#	Name	Member Type
1	SASGOPT	CATALOG
2	SASMAC3	CATALOG
3	SASMACR	CATALOG

Connecting the CDISC CORE Rules Engine

Connecting the CDISC CORE Rules Engine



https://wmt-001.my.workbench.sas.com/sas-workbench-4863tpwtwgonyku2gesgaj094/?folder=/workspaces/cdisc_c...
cdisc_core_2
EXPLORER
CDISC_CORE_2
README.md
[Preview] README.md
OUTLINE
TIMELINE
Layout: Swiss German

Welcome to SAS® Viya® Workbench

Looking for examples to get you started?

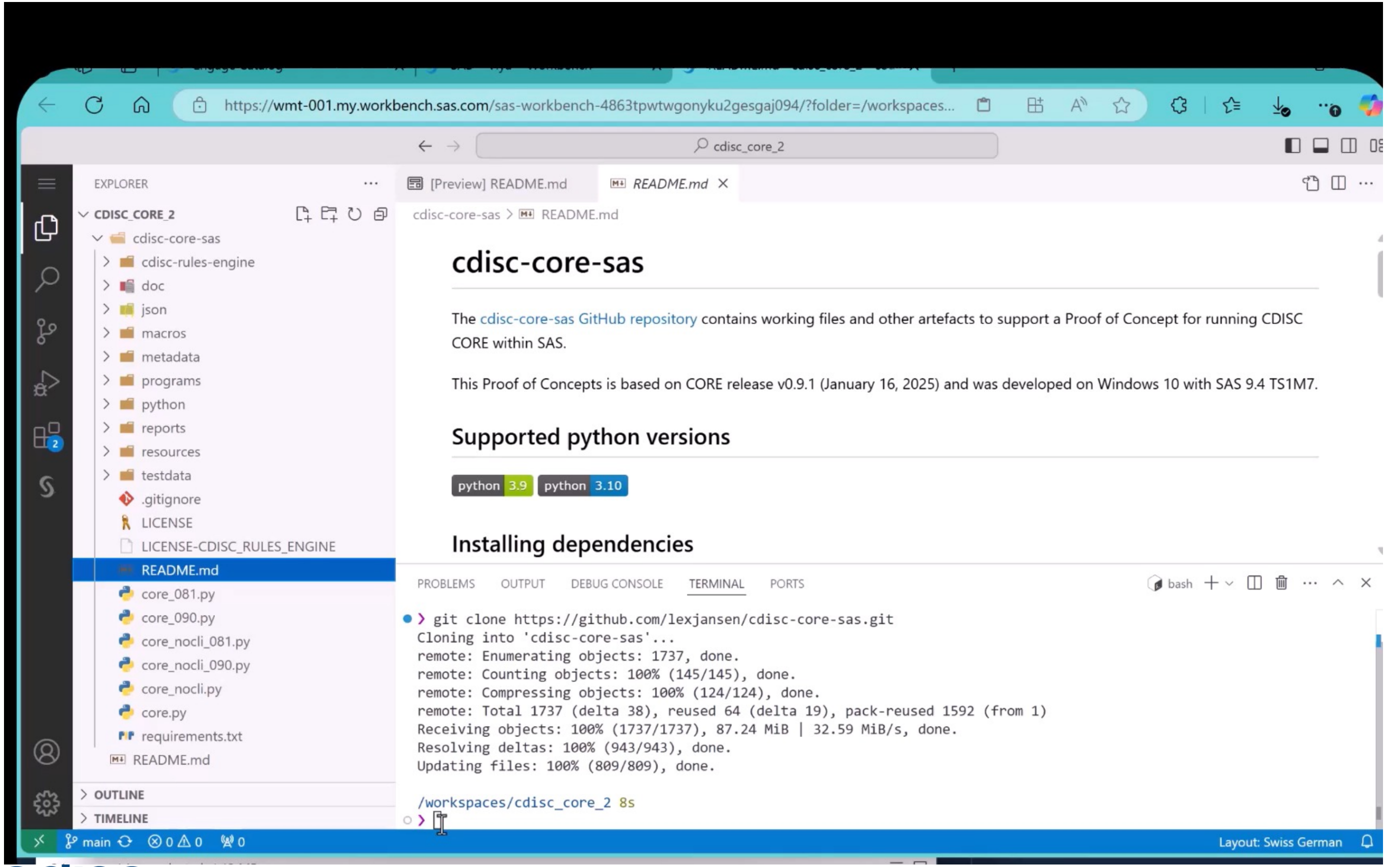
SAS and Python code examples for use with SAS® Viya® Workbench are available on GitHub. After starting SAS® Viya® Workbench, open a terminal* and run the following command to clone the examples repository into the root of your workspace:

```
git -C $WORKSPACE clone https://github.com/sassoftware/sas-viya-workbench-examples.git
```

* To open a terminal, you can use the key combination `Control` + `` ` `` or enter the menu at the top left and navigate to `Terminal` followed by `New Terminal`.

Setting up the Python environment for the CDISC CORE Rules engine

Setting up the Python environment



https://wmt-001.my.workbench.sas.com/sas-workbench-4863tpwtwgonyku2gesgaj094/?folder=/workspaces...

cdisc-core-sas

The cdisc-core-sas GitHub repository contains working files and other artefacts to support a Proof of Concept for running CDISC CORE within SAS.

This Proof of Concepts is based on CORE release v0.9.1 (January 16, 2025) and was developed on Windows 10 with SAS 9.4 TS1M7.

Supported python versions

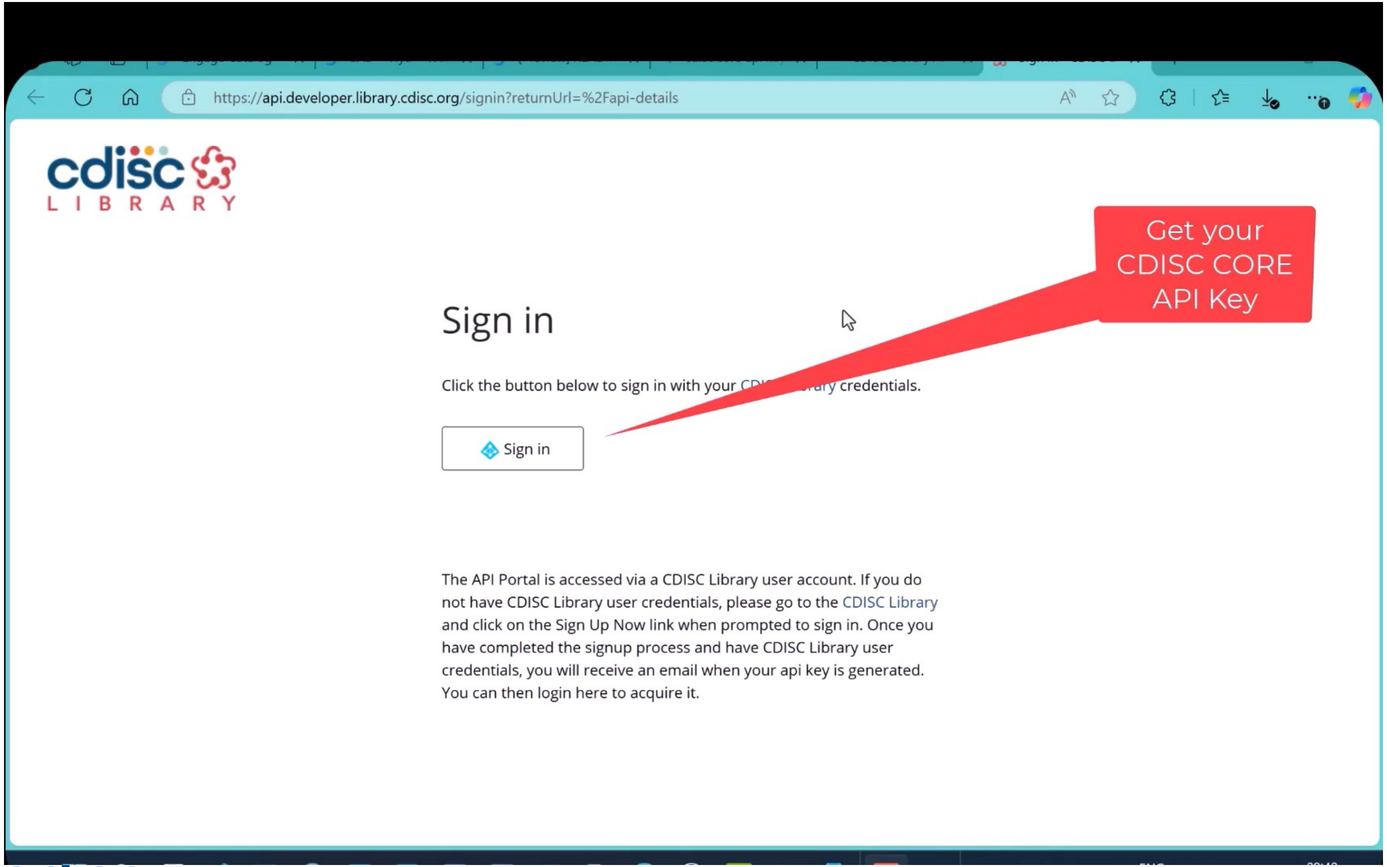
python 3.9 python 3.10

Installing dependencies

```
> git clone https://github.com/lexjansen/cdisc-core-sas.git
Cloning into 'cdisc-core-sas'...
remote: Enumerating objects: 1737, done.
remote: Counting objects: 100% (145/145), done.
remote: Compressing objects: 100% (124/124), done.
remote: Total 1737 (delta 38), reused 64 (delta 19), pack-reused 1592 (from 1)
Receiving objects: 100% (1737/1737), 87.24 MiB | 32.59 MiB/s, done.
Resolving deltas: 100% (943/943), done.
Updating files: 100% (809/809), done.

/workspaces/cdisc_core_2 8s
>
```

Executing the CDISC CORE Rules Engine

Executing the CDISC CORE Rules Engine



https://api.developer.library.cdisc.org/signin?returnUrl=%2Fapi-details
cdisc LIBRARY

Sign in

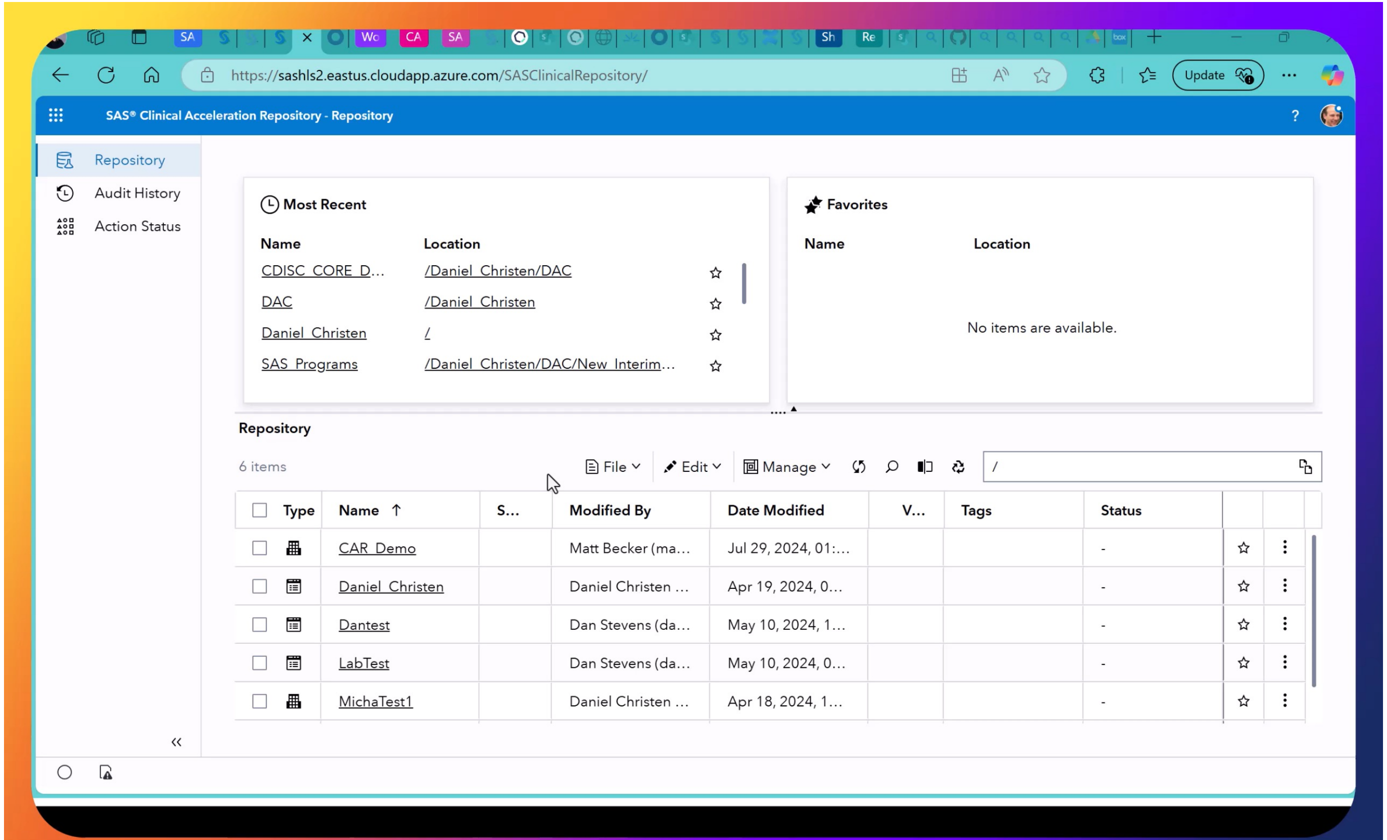
Click the button below to sign in with your CDI[illegible]rary credentials.

The API Portal is accessed via a CDISC Library user account. If you do not have CDISC Library user credentials, please go to the CDISC Library and click on the Sign Up Now link when prompted to sign in. Once you have completed the signup process and have CDISC Library user credentials, you will receive an email when your api key is generated. You can then login here to acquire it.

Get your CDISC CORE API Key

Writing your final report to the repository under version control

Writing the final report to your repository



https://sashls2.eastus.cloudapp.azure.com/SASClinicalRepository/

SAS® Clinical Acceleration Repository - Repository

Most Recent

Name	Location
CDISC_CORE_D...	/Daniel_Christen/DAC
DAC	/Daniel_Christen
Daniel_Christen	/
SAS_Programs	/Daniel_Christen/DAC/New_Interim...

Favorites

Name	Location

No items are available.

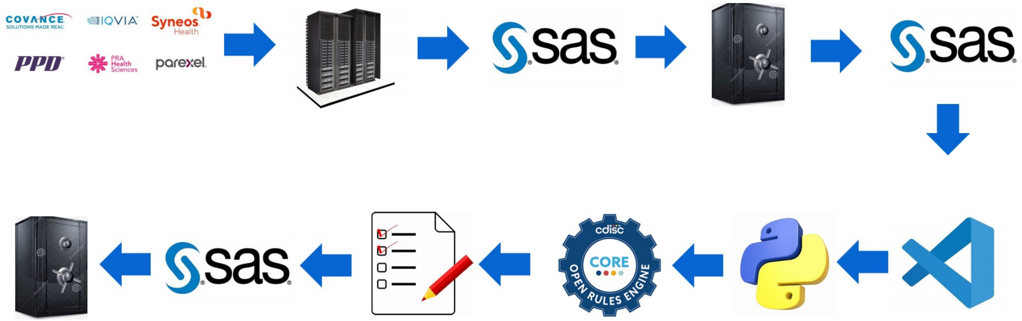
Repository

6 items

Type	Name ↑	S...	Modified By	Date Modified	V...	Tags	Status
	CAR_Demo		Matt Becker (ma...	Jul 29, 2024, 01:...			-
	Daniel_Christen		Daniel Christen ...	Apr 19, 2024, 0...			-
	Dantest		Dan Stevens (da...	May 10, 2024, 1...			-
	LabTest		Dan Stevens (da...	May 10, 2024, 0...			-
	MichaTest1		Daniel Christen ...	Apr 18, 2024, 1...			-

Summary

Summary



By combining the CDISC CORE Rules Engine, VS Code, SAS, Git, and Python capabilities, we can efficiently validate clinical trial data against CDISC standards and securely store the final report under version control in your preferred clinical data repository.

Q&A

Thank You!