

GUIDE

AtScale Developer Edition Quick Start Guide

ATSCALE

Table of Contents

| | |
|---|----|
| Minimum Requirements | 1 |
| Install Docker | 1 |
| Setting up Docker Desktop Resource Limits | 1 |
| Install AtScale Server | 2 |
| What is AtScale Design Center? | 4 |
| Quick Start: Internet Sales Tutorial | 5 |
| Deploy AtScale Tutorial Model | 6 |
| Query AtScale Tutorial Model Using a BI Tool | 6 |
| Connect Tableau | 7 |
| Connect Power BI | 8 |
| Connect Excel | 9 |
| Create a Private Repository | 11 |
| Set up a Git Repository for AtScale's Semantic Models Library | 11 |
| Add Your Own Repository to AtScale | 12 |

Minimum Requirements

The following minimum requirements are suggested for running AtScale:

1. Chrome Browser
2. Docker configured with 8 Cores and 8GBs of RAM
3. 64GBs of available disk

Install Docker

Before you start this guide, you must have Docker Desktop installed on your local computer. Please follow these instructions:

 [Install Docker Desktop on Windows](#)

 [Install Docker Desktop on Mac](#)

 [Install Docker Desktop on Linux](#)

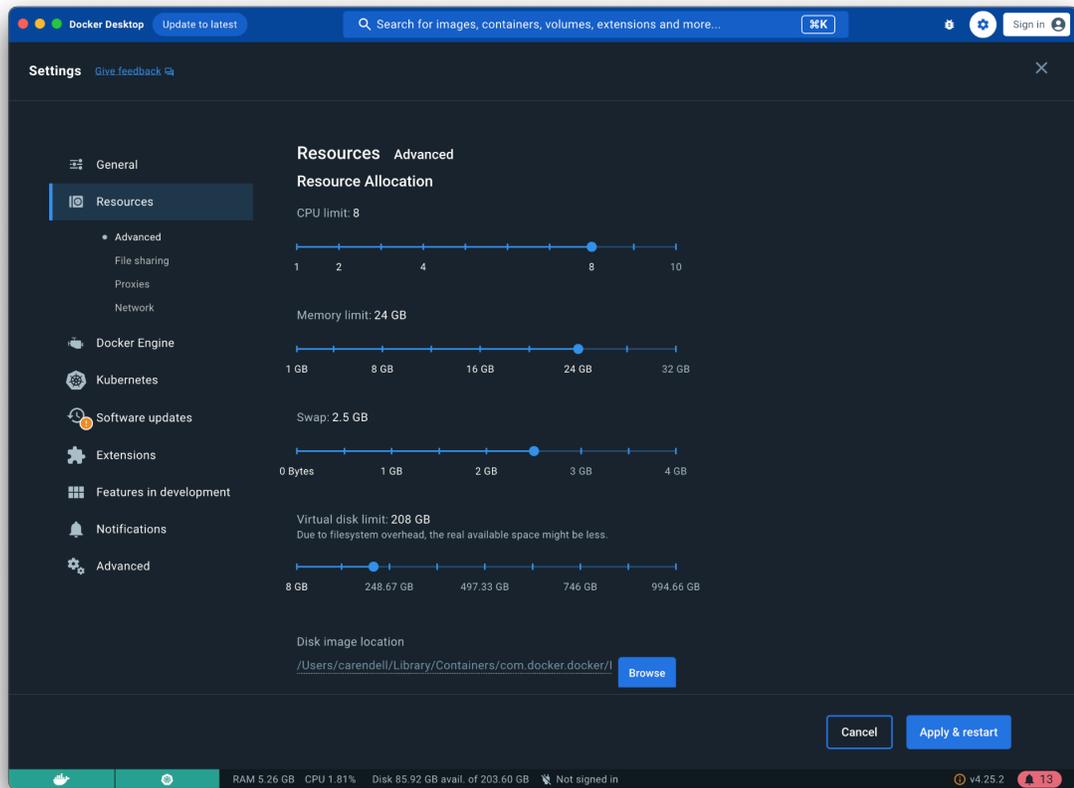
Setting up Docker Desktop Resource Limits

Once Docker Desktop is installed:

1. Click on the cog wheel in the top right-hand corner of the window.



2. Click on **Resources** on the left sidebar to show the resource settings.
 - a. Increase the CPU limit to at least 6 cores, up to 8 cores for improved performance.
 - b. Increase the memory limit to at least 8GB, up to 16GB for improved performance.
 - c. Increase the virtual disk limit to 128GB and up to 256GB.



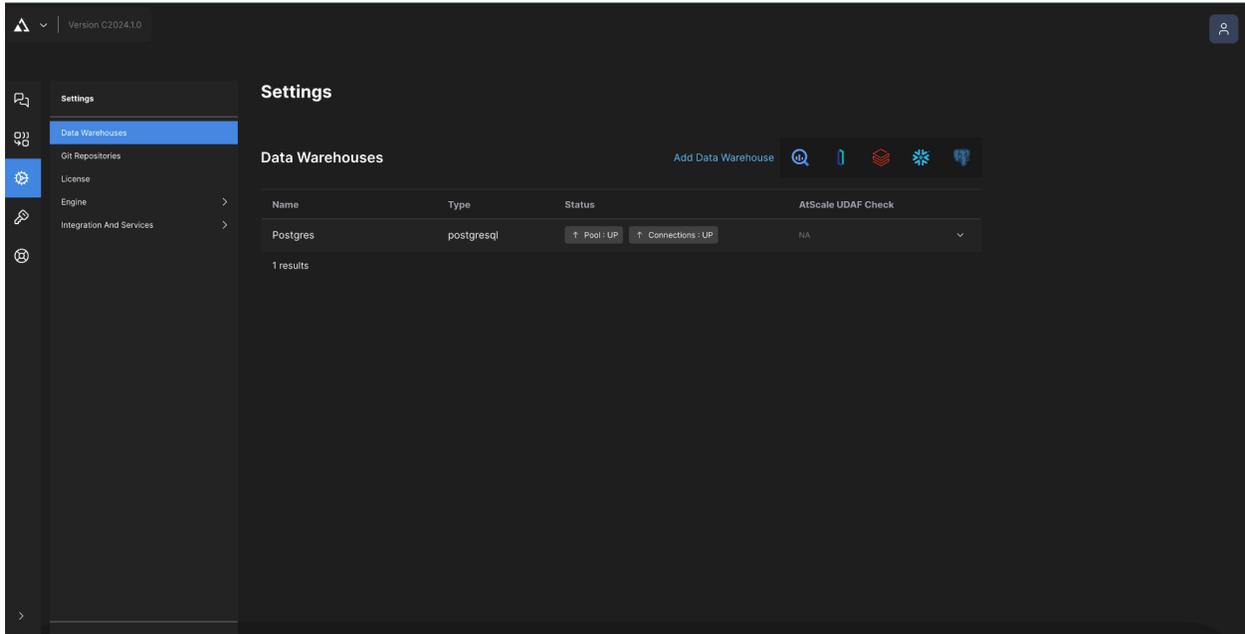
Install AtScale Server

1. Download the file sent to your email.
 - a. It should be: devops-community-stack-develop.zip
2. Unzip the file.
3. Open a **Terminal**.
4. Navigate to the folder you've just unzipped and into compose.
 - a. "cd Downloads/devops-community-stack-develop/compose/"
5. Now, run the following command: `docker compose up -d`

```
C:\Users\ user \Desktop\docker\Fresh Community\devops-community-stack\compose>docker compose up -d
```

6. Wait for the containers to come up successfully.
 - a. **Note:** *Even though the containers are successfully downloaded and started, all the AtScale services may take some time to become available. After logging into AtScale Design Center and applying your license key, check the Postgres tutorial database's status by clicking*

on the "Data Warehouse" menu in "Settings" and make sure that the Pool Status and Connection Status show "Up" as illustrated below:



```
C:\Windows\System32\cmd.exe

promtail Pulled 2.5s
entitlement Pulled 2.5s
kong Pulled 2.6s
loki Pulled 2.5s
postgres Pulled 2.5s
collector Pulled 2.5s
api Pulled 2.6s
identity Pulled 2.6s
[+] Running 13/19
- Network atscale-community-edition_default Created 76.3s
- Volume "atscale-community-edition_atscale-tutorial" C... 76.1s
- Volume "atscale-community-edition_atscale-tempo" Crea... 76.1s
- Volume "atscale-community-edition_atscale-postgres" C... 76.1s
- Volume "atscale-community-edition_atscale-loki" Creat... 76.1s
- Volume "atscale-community-edition_atscale-cache" Crea... 76.1s
Container atscale-community-edition-redis-1 Healthy 17.4s
Container atscale-community-edition-postgres-1 Health... 7.1s
Container atscale-community-edition-collector-1 Start... 2.2s
Container atscale-community-edition-tempo-1 Started 1.6s
Container atscale-community-edition-promtail-1 Starte... 2.2s
Container atscale-community-edition-loki-1 Started 1.0s
Container atscale-community-edition-kong-1 Started 1.6s
Container atscale-community-edition-identity-1 Health... 47.8s
Container atscale-community-edition-entitlement-1 Hea... 58.8s
Container atscale-community-edition-web-1 Started 48.3s
Container atscale-community-edition-api-1 Healthy 69.6s
Container atscale-community-edition-tutorial-1 Health... 75.4s
Container atscale-community-edition-engine-1 Started 75.6s

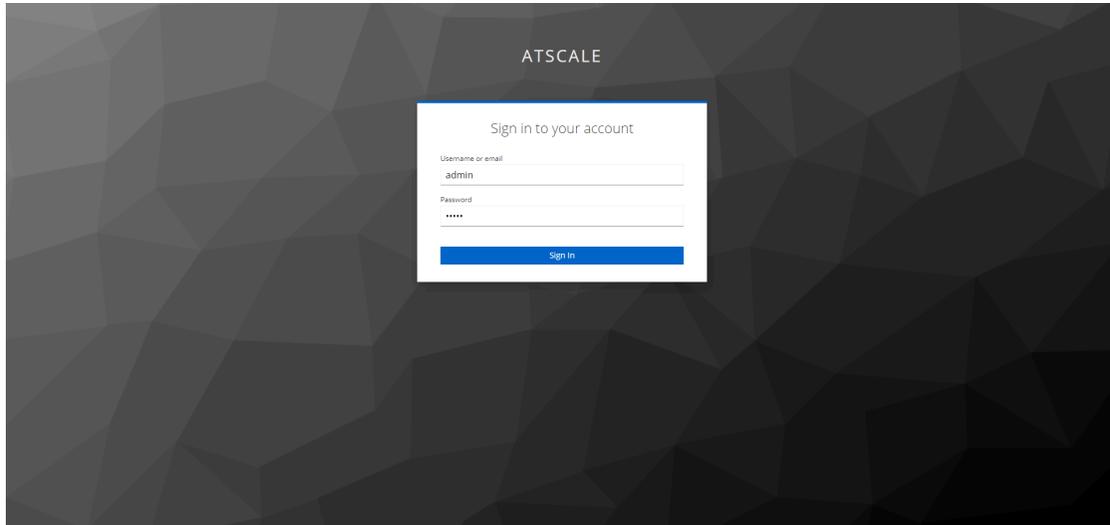
C:\Users\Petar Staykov\Desktop\Docker\Fresh Community\devops-community-stack\compose>
```

7. Point your browser to <http://localhost>

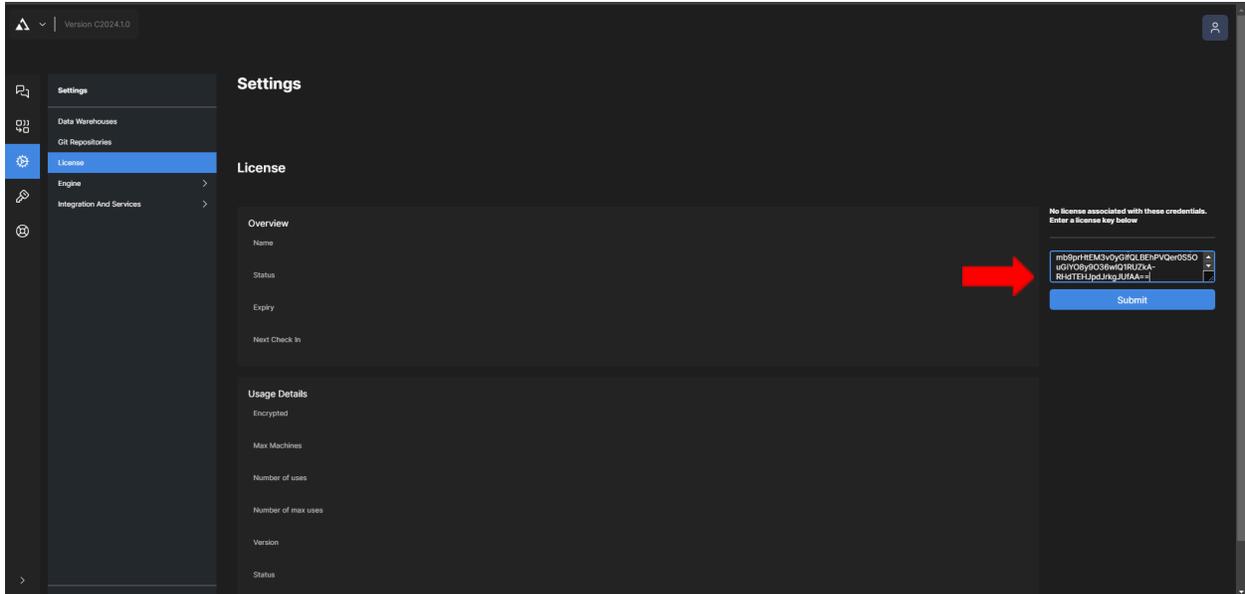
What is AtScale Design Center?

AtScale Design Center is a semantic modeling tool that helps business analysts and analytics engineers build semantic models. AtScale Semantic Models are based on the Semantic Modeling Language (SML) syntax. Please see the [Semantic Modeling Guide](#) for more information.

1. To start, log into the [Design Center](#).



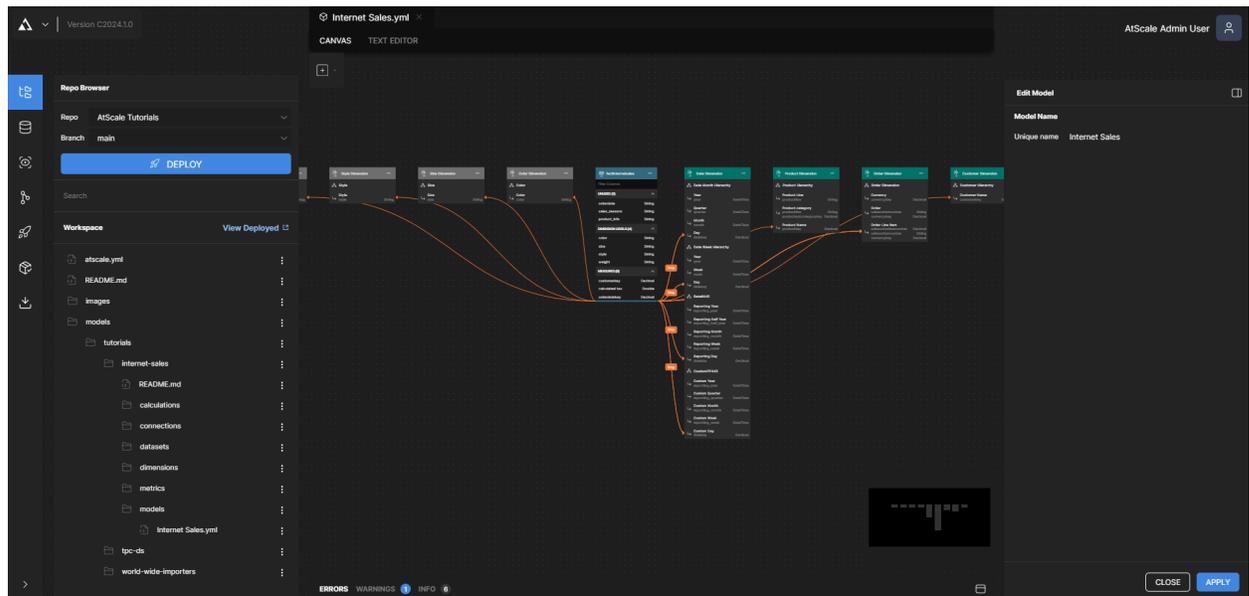
- a. Use the username **admin**.
 - b. Use the password **admin**.
2. For security reasons, AtScale will ask you to update your admin password after you first login.
 3. Next, input your AtScale license and hit **Submit**.
 - a. **Note:** Your unique license key and the download link were emailed directly to you from community@atscale.com.



4. Give AtScale some time to finalize the setup.
 - a. **Note:** It should be about 3 minutes.

Quick Start: Internet Sales Tutorial

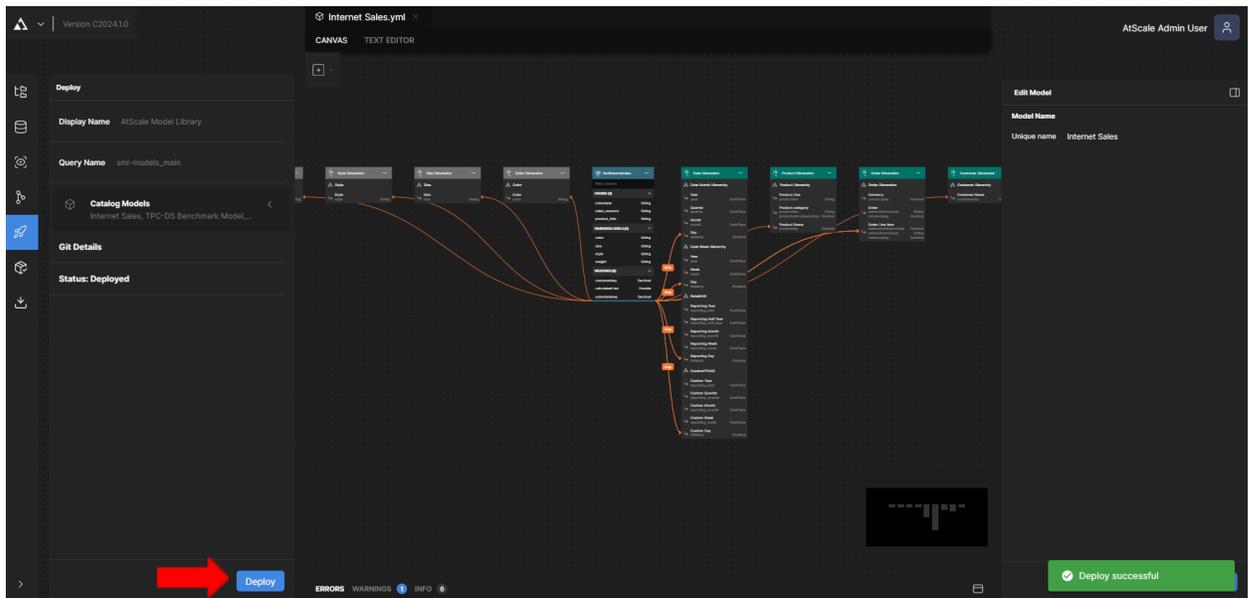
1. Navigate to **Workspace** → **Repo Browser** → **Models** → **Internet Sales** → **Models** → **Internet Sales.yml**.



Deploy AtScale Tutorial Model

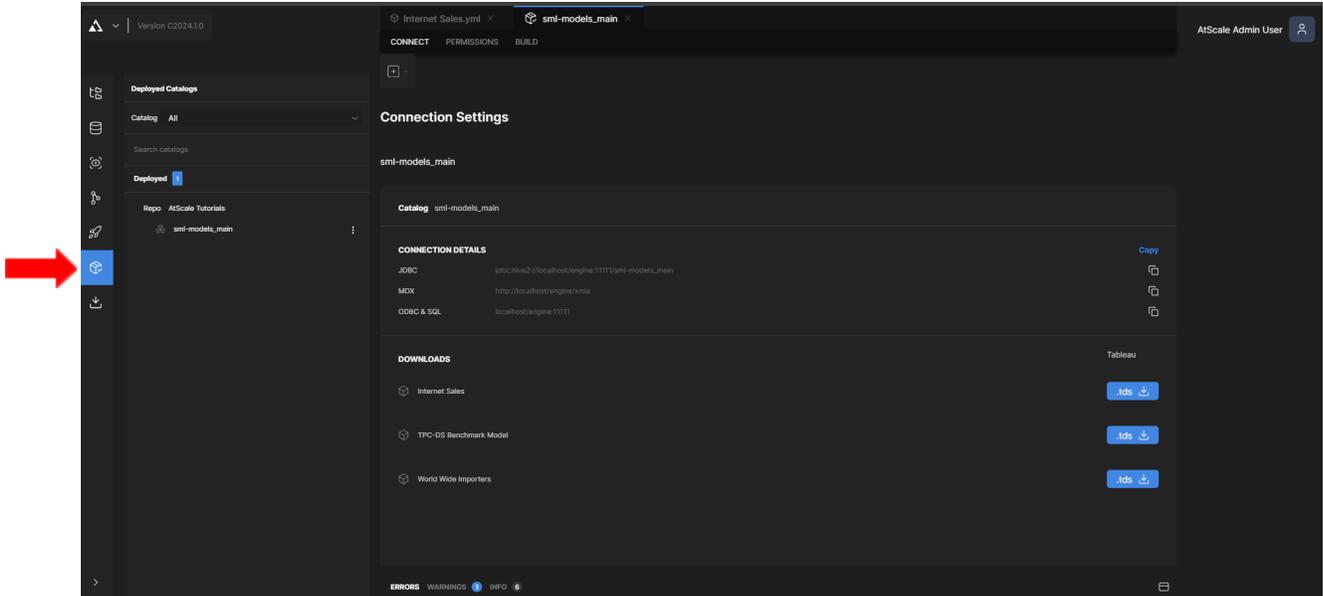
To explore and query the tutorial models in the AtScale Design Center, deploy your catalog.

Click the **Repo Browser** icon in the Activity bar (left side) and press the **Deploy** button.



Query AtScale Tutorial Model Using a BI Tool

Click on the Deployed Catalogs icon in the Activity bar (left side) and click on your deployed catalog. Instructions for connecting your BI tools will appear on the right side.



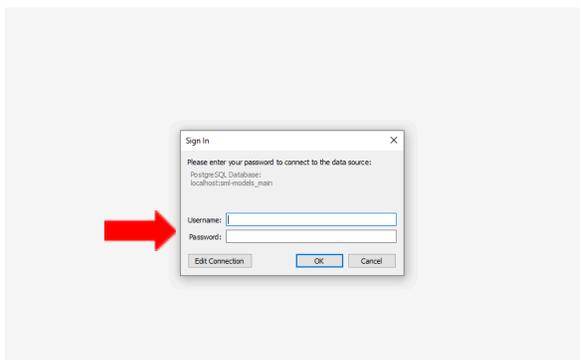
Special Note When Connecting Tools From Other Machines/VMs

Since the Developer Edition is installed by default as “localhost”, if you wish to connect to AtScale from locations other than your host machine, you must replace “localhost” in the AtScale connection strings with a real IP address. In Windows, you can find your IP address by running the “ipconfig” command in the terminal or by running the “ifconfig” command in a Linux/Mac terminal.

Connect Tableau

Before you start:

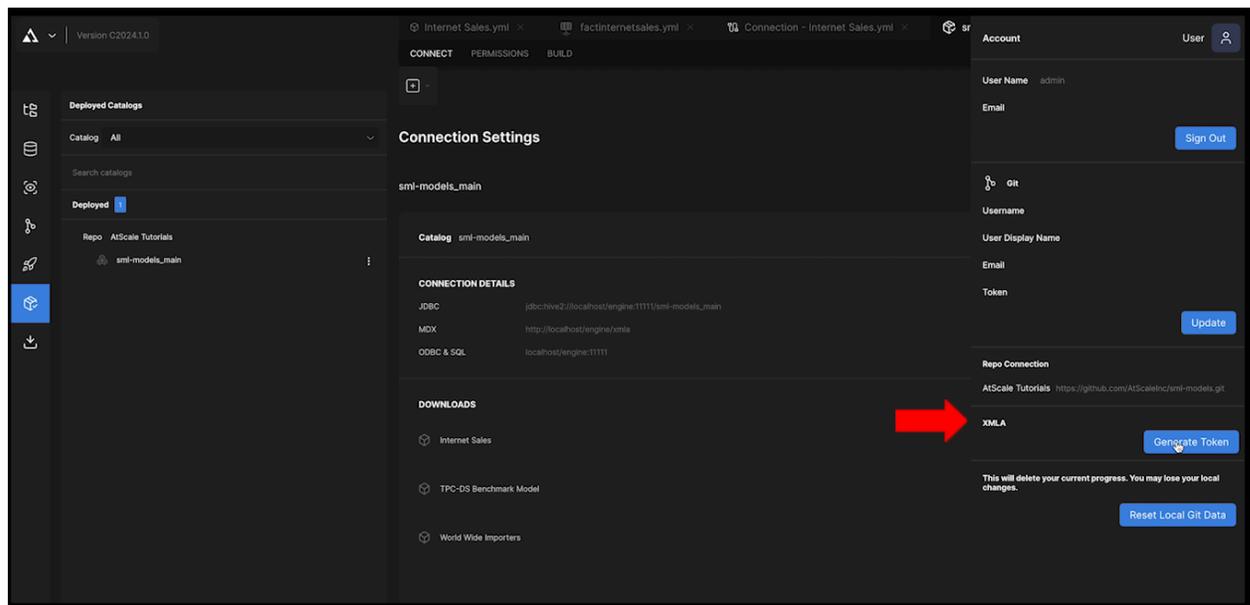
1. Download the Java 8 JDBC driver from <https://jdbc.postgresql.org/download/>
2. Copy the .jar file to the following folder (you may have to create the folder manually).
3. Download and double-click the TDS file.



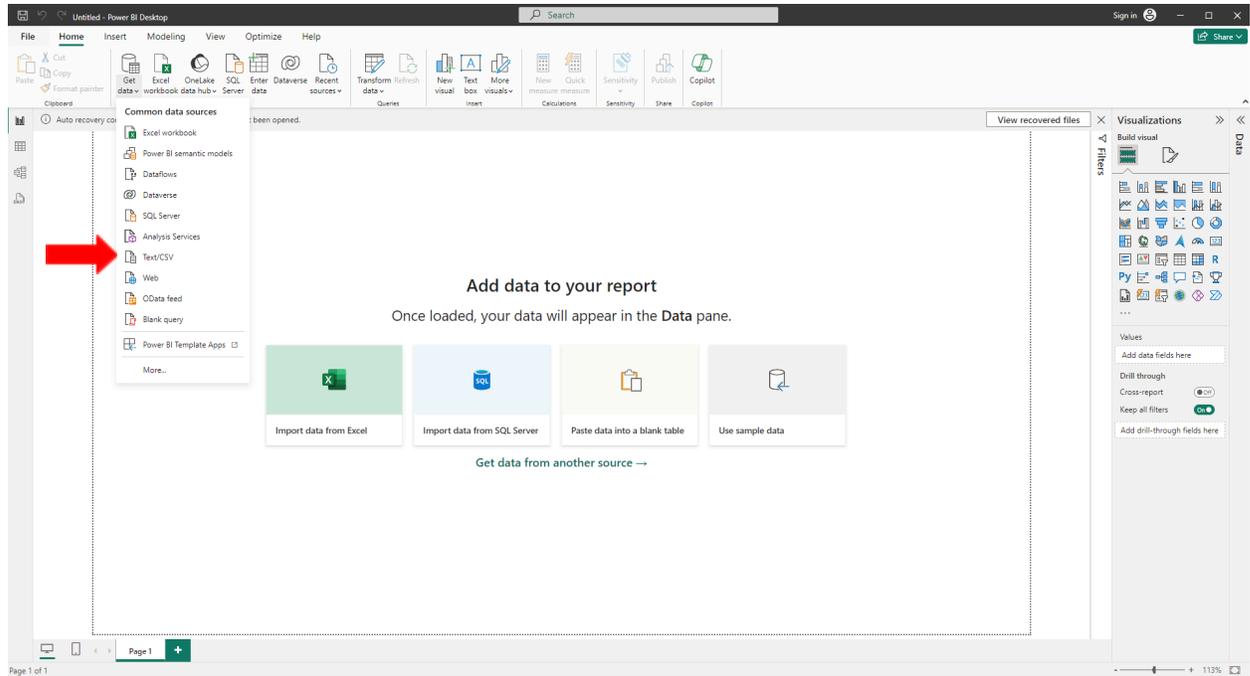
4. Enter the user credentials.
 - a. Use the username **admin**.
 - b. Enter your password:
 - i. Use the default password **admin** or the one you changed to upon the first login in the Design Center.

Connect Power BI

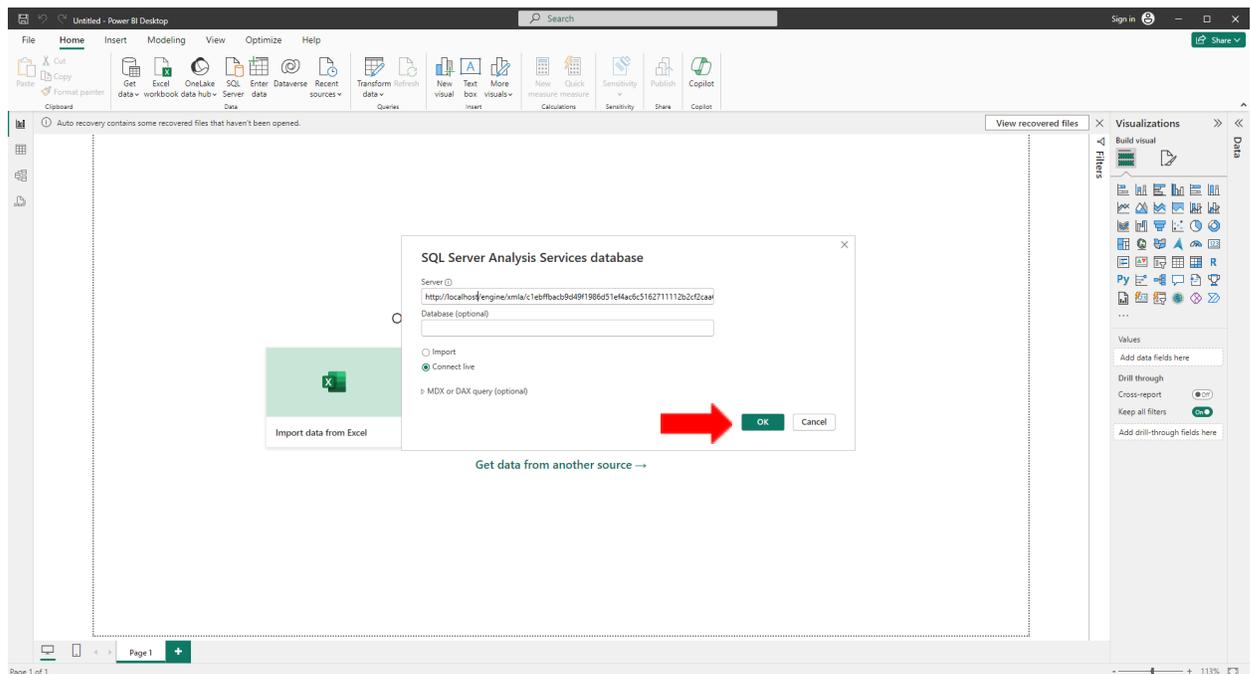
1. Under the Design Center user profile, click **Generate XMLA token**.



2. Copy the URL from MDX + Token.
3. In PBI, go to **Get Data** → **Analysis Services**.

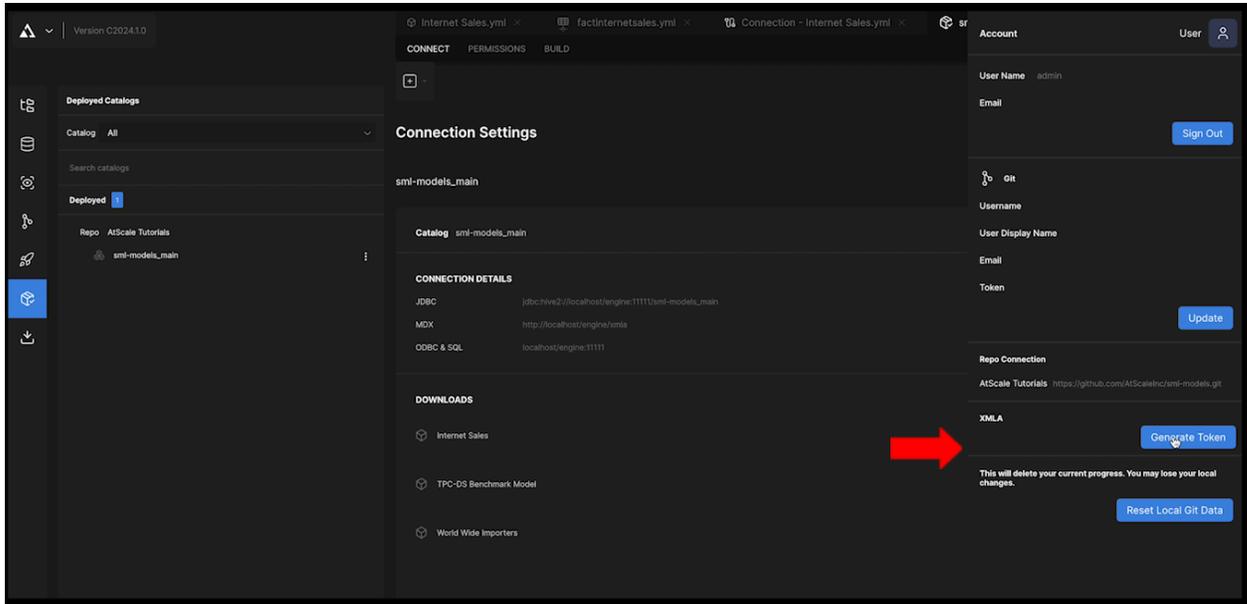


4. Paste the URL you obtained in the previous step and click **OK**.

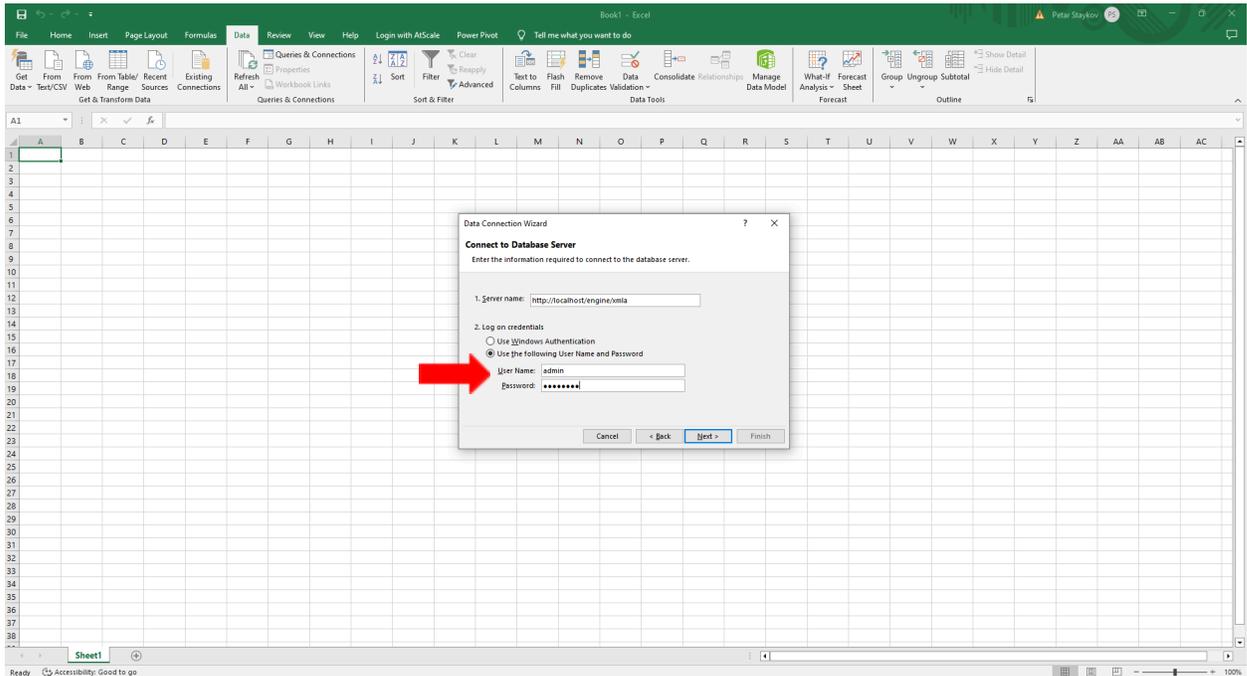


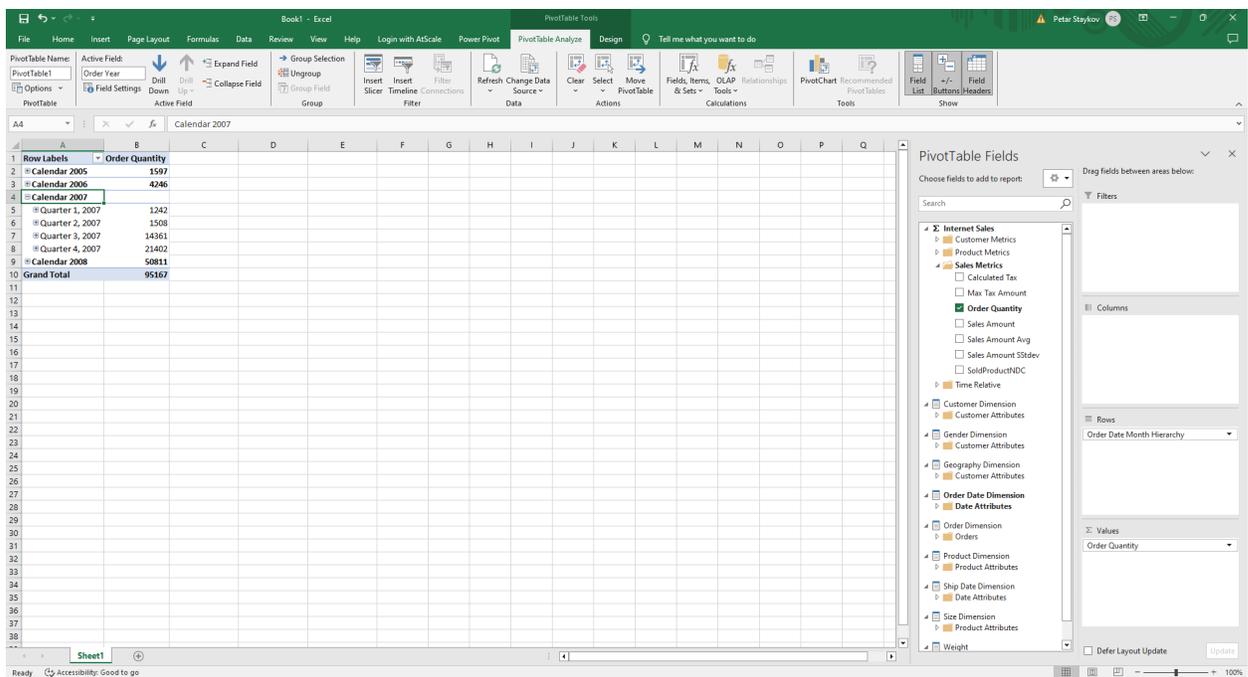
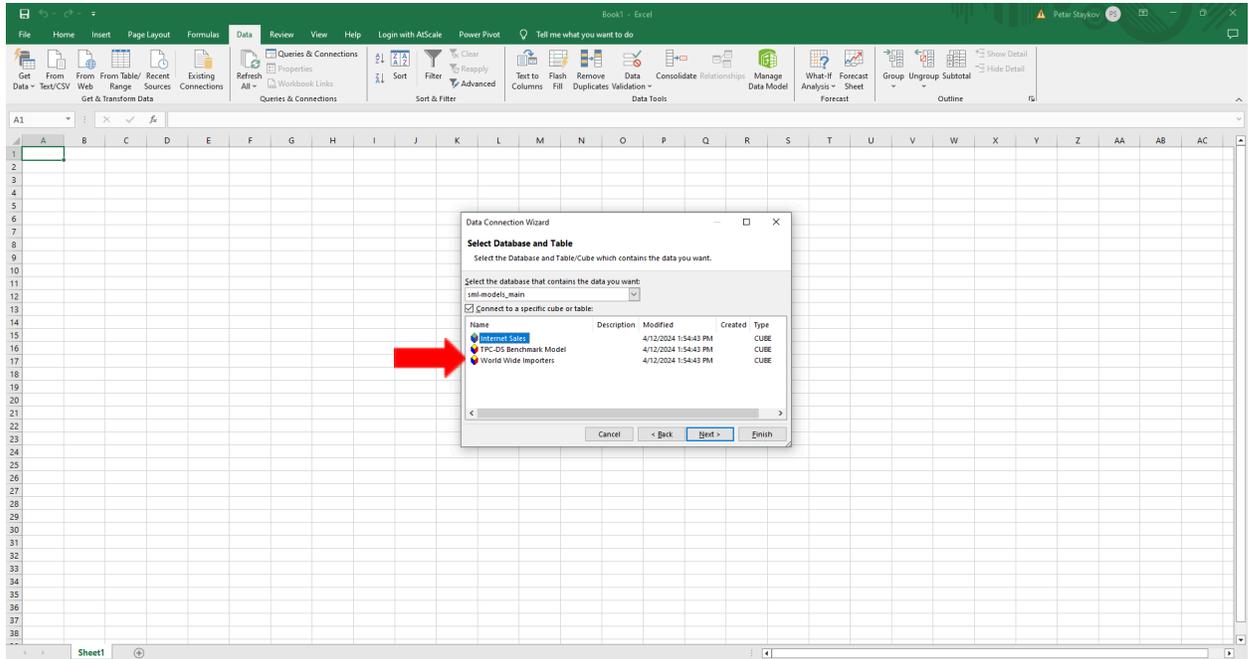
Connect Excel

1. Under the Design Center user profile, click **Generate XMLA token**.



2. Copy the MDX connection string from the connection page in the Design Center.
3. In Excel, go to **Data**→**Get Data**→**From Database**→**From Analysis Services**.





Create a Private Repository

Set up a Git Repository for AtScale's Semantic Models Library

1. In Git, click on **New Repository**.

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)

Required fields are marked with an asterisk (*).

[Single sign-on](#) to see results in the AtScaleInc organization.

Owner * / Repository name *

Great repository names are short and memorable. Need inspiration? How about [studious-engine](#) ?

Description (optional)

[Please choose an owner](#) to see the available visibility options.

Initialize this repository with:

Add a README file
This is where you can write a long description for your project. [Learn more about READMEs.](#)

Add .gitignore

.gitignore template:

Choose which files not to track from a list of templates. [Learn more about ignoring files.](#)

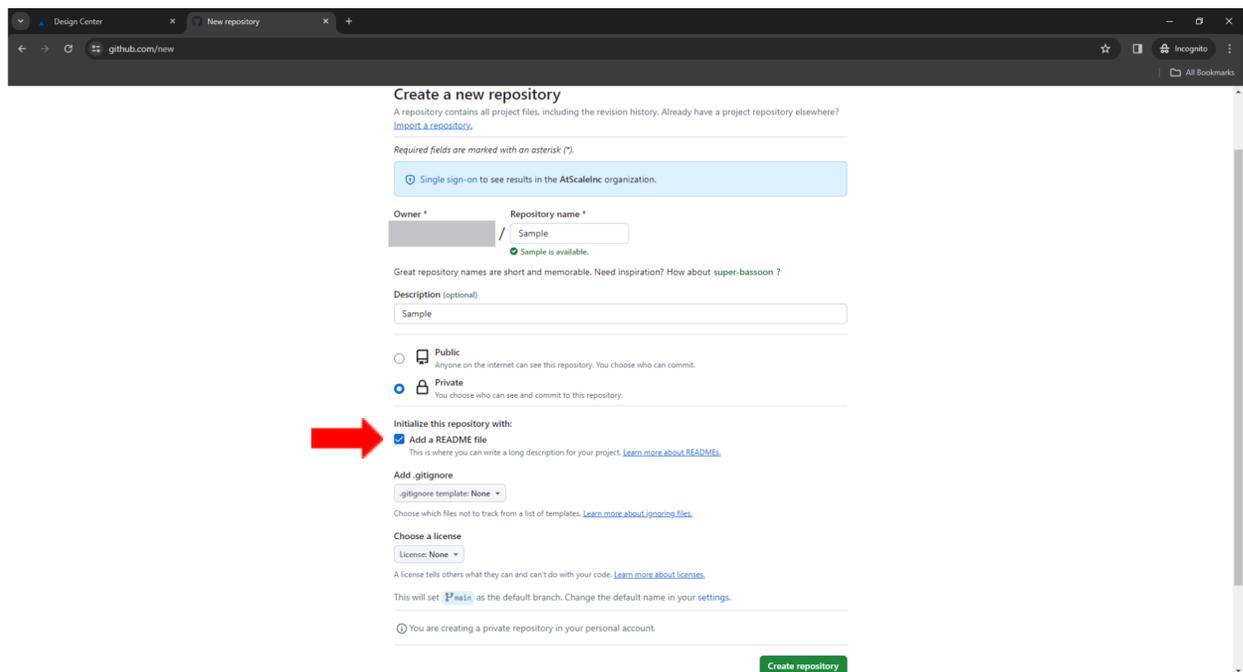
Choose a license

License:

A license tells others what they can and can't do with your code. [Learn more about licenses.](#)

[Create repository](#)

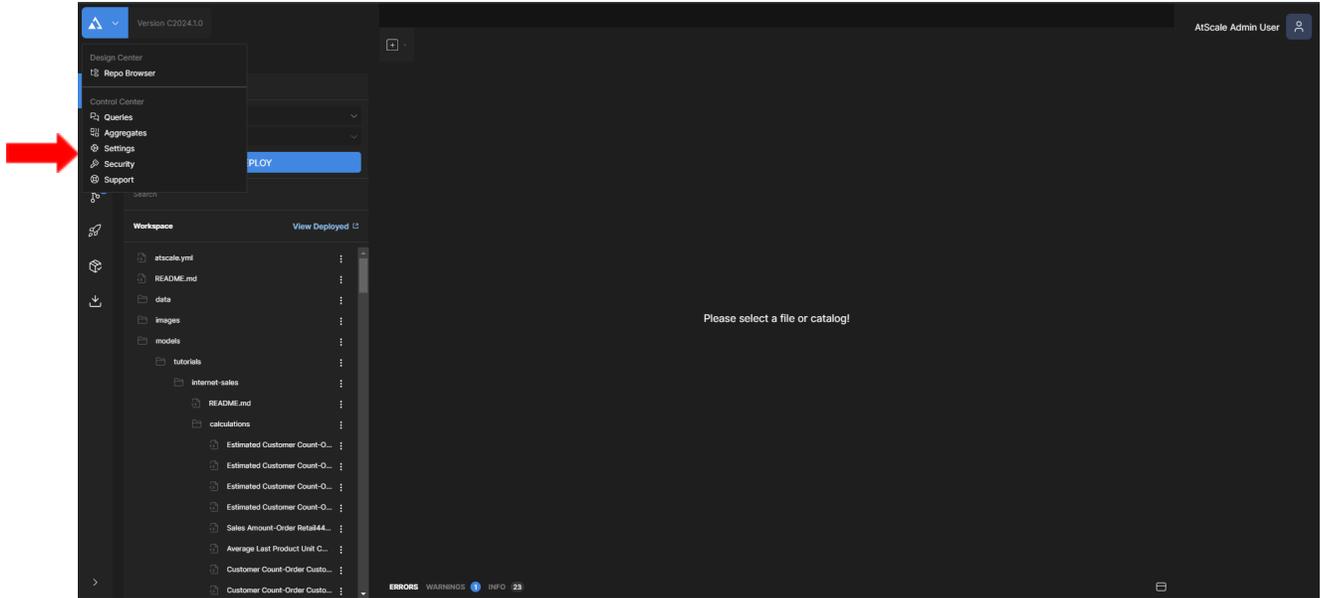
2. Add name and description and check the **Add a Readme file** checkbox.



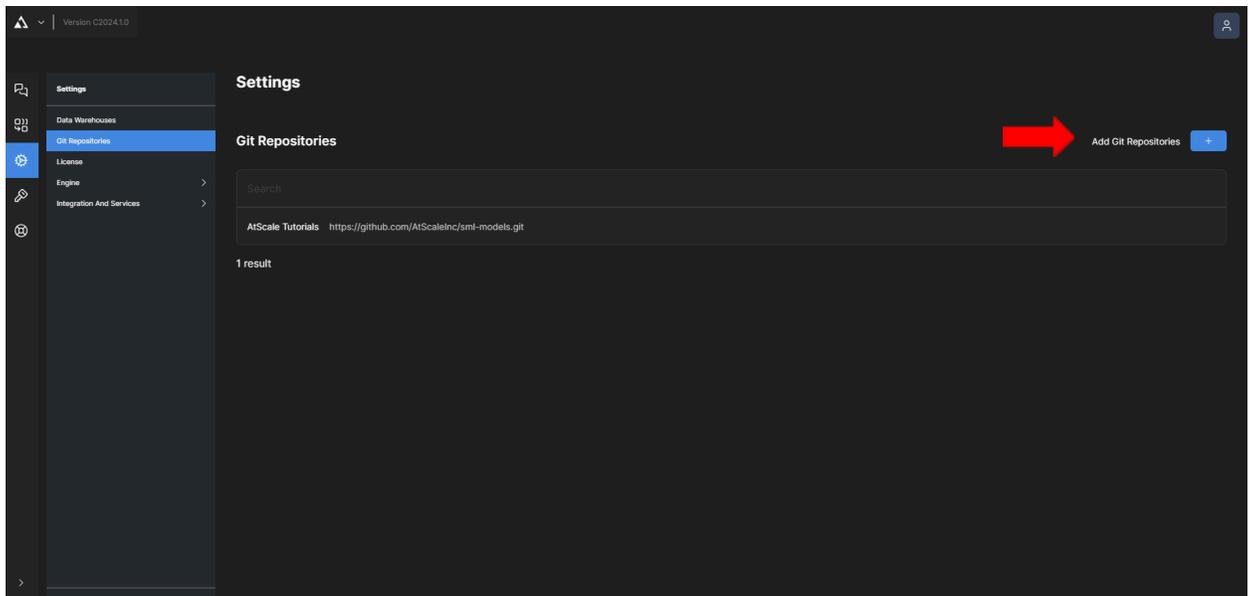
3. Copy the HTTPS URL.

Add Your Own Repository to AtScale

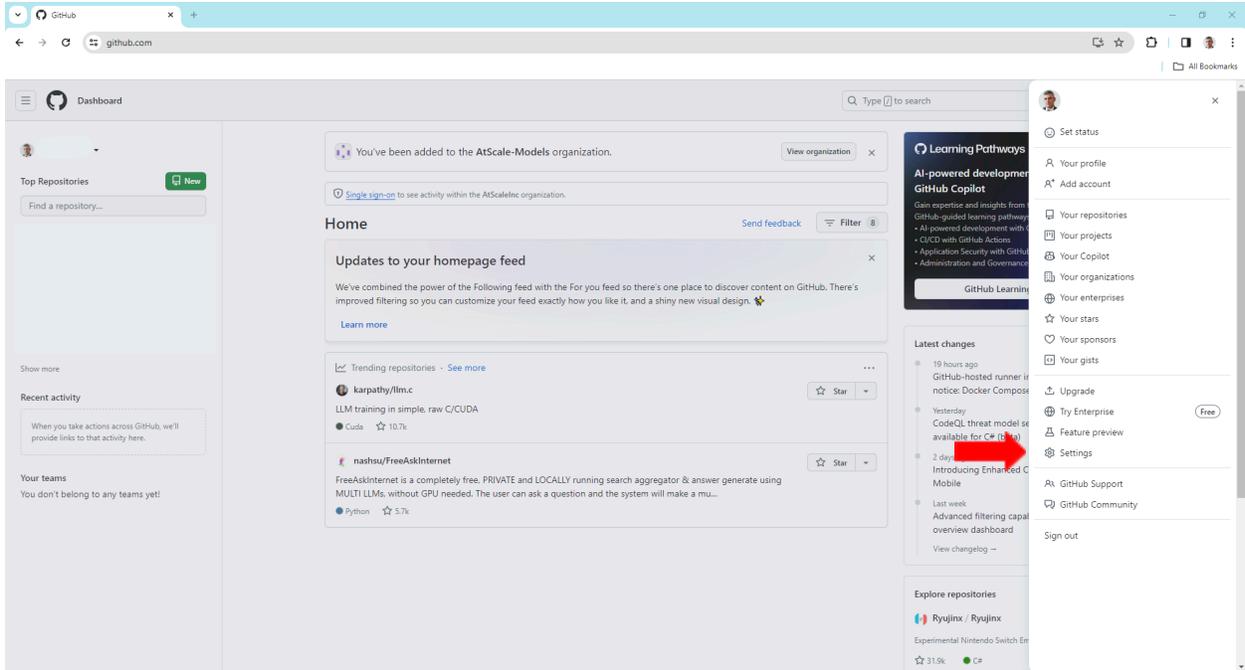
1. Go to AtScale Settings: In the AtScale Design Center, click on the **Settings** dropdown by clicking on the AtScale logo icon (upper left).



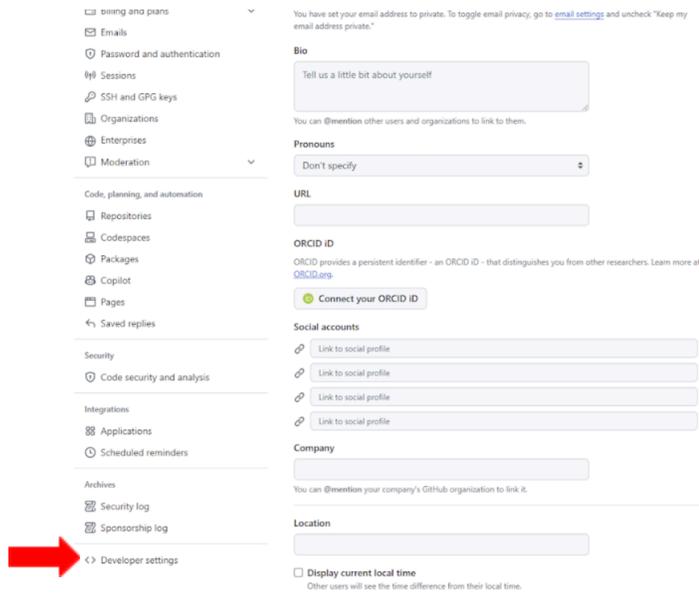
2. In the Git Repositories section, click **Add**.



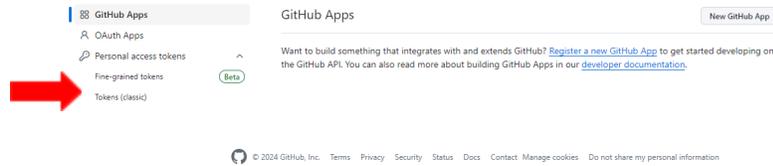
3. Click **Save** and the new repo will appear in the dropdown menu in the repo browser.
4. Login to your Git Account.
5. Under your User Profile, click→ **Settings**.



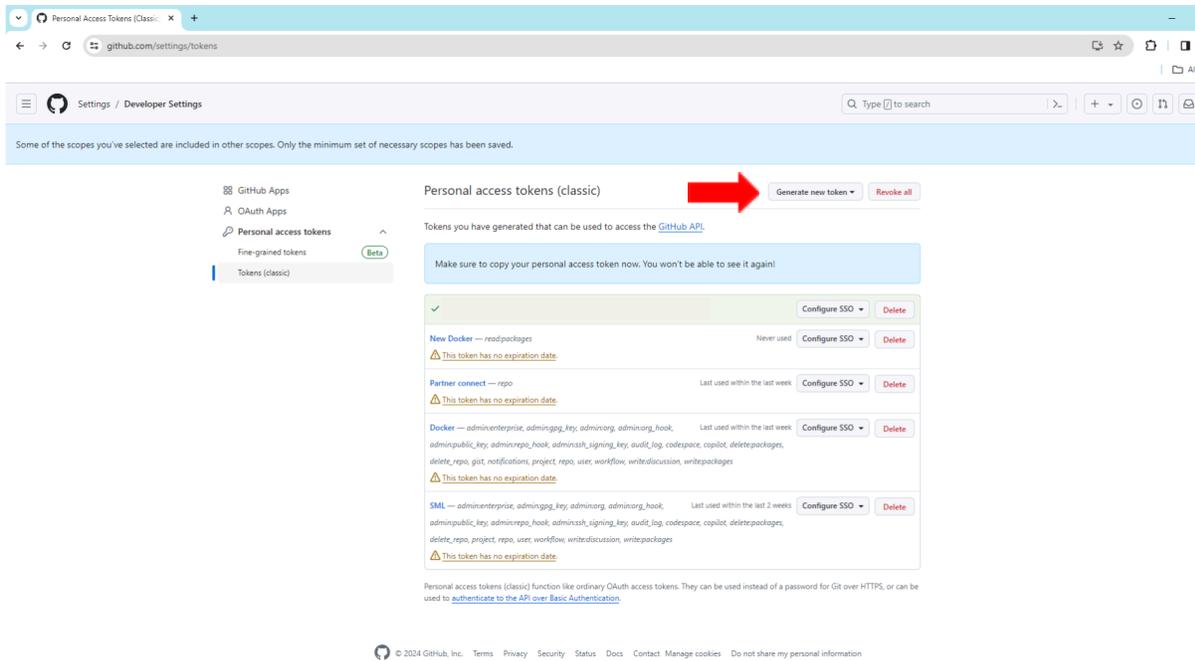
6. Click **Developer Settings**.



7. Click **Personal Access Token** → **Token Classic**.



8. Click **Generate**. Give it a name and check the repo rights.

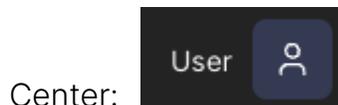


9. At this stage, copy the token and save it somewhere else. You will need to use this token in the AtScale Design Center.

- a. **Note:** *This is the last time GitHub will show you the token. Save the token somewhere safe that you can easily access. If you forget or lose your token, go back to step 5 and repeat. There is no way to recover a token.*

10. Enter your Git Credentials into the AtScale Design Center.

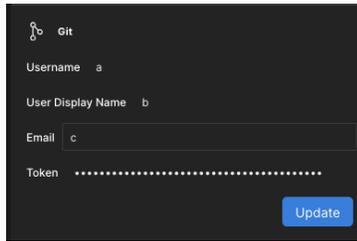
- a. Click the Profile Icon in the upper right-hand corner of AtScale Design



Center:

- b. Fill in the Git Details. The fields Username, User Display Name, and Email are all MANDATORY, but you can put whatever you want in them. They are not validated.

c. Put the token from GitHub into the Token Field.



A screenshot of the Git configuration interface. It features a dark background with white text. At the top left is the Git logo and the word "Git". Below it are four input fields: "Username" with the value "a", "User Display Name" with the value "b", "Email" with the value "c", and "Token" with a series of dots. A blue "Update" button is located at the bottom right of the form.

11. Upon creating your first model the atscale.yml will be generated automatically by the system.

