# ADRF Onboarding Handbook

Josh Edelmann and Corey Sparks

Last Updated on 11 August, 2023

# Table of contents

# Preface

This is a revised version of the Coleridge Initiative ADRF User On-boarding Handbook

This is a living document intended to show new ADRF users how to use the platform for common tasks

# 1 Obtaining ADRF Access and Account Set Up

## 1.1 Requesting an Account

- Agency-affiliated researcher. If you are an agency-affiliated researcher, your agency will set up an ADRF account for you.
- Individual part of a training program. If you are part of a training program, Coleridge Initiative will create an account for you once you have been accepted into the program.

## 1.2 Account Registration and Onboarding Tasks

- You will receive an email invitation to activate your account. The email will come from http://okta.com, so please make sure that it doesn't get caught in your email spam filter. Follow the steps outlined in the email to set up your password and your multi-factor authentication preferences. Clink on the link below to watch a video walking through the steps.
- After activating your account, you will be logged in to the ADRF Applications page. Proceed to the Data Stewardship application by clicking on the icon.
- In the Data Stewardship application, you will notice a "Tasks" section with a number of items you will need to complete before you can gain access to the project space. Refer to the next section for details about the onboarding process.

## 1.3 Obtaining ADRF Access

- Agency-affiliated researcher. If you are an agency-affiliated researcher using an agency-sponsored account, you will be granted ADRF access once you complete your onboarding tasks and required data access agreements. If you are an self-paying agency-affiliated researcher, your ADRF access is conditional on receipt of payment. If your institution of Office of Sponsored Programs will be submitting payment on your behalf, please be aware of potential access delays. Whenever possible, the Coleridge Initiative advises paying with a personal credit card or institutional payment card and using the generated invoice to request reimbursement.

- Individual part of a training program. If you are part of a training program, you will be granted ADRF access once you complete your onboarding tasks and required data access agreements.

## 1.4 More Information

If you have any questions, please contact support@coleridgeinitiative.org.

# 2 Onboarding Modules and Security Training

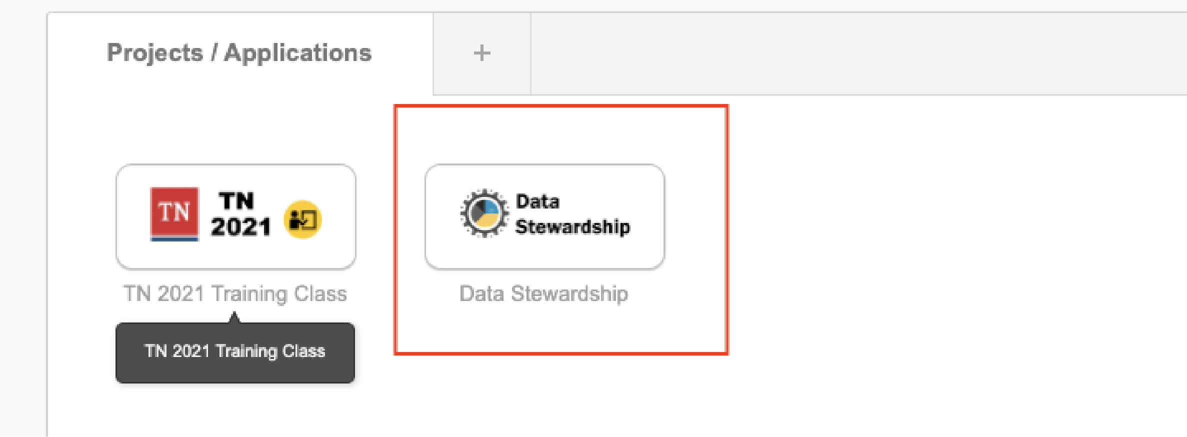
## 2.1 Data Stewardship App

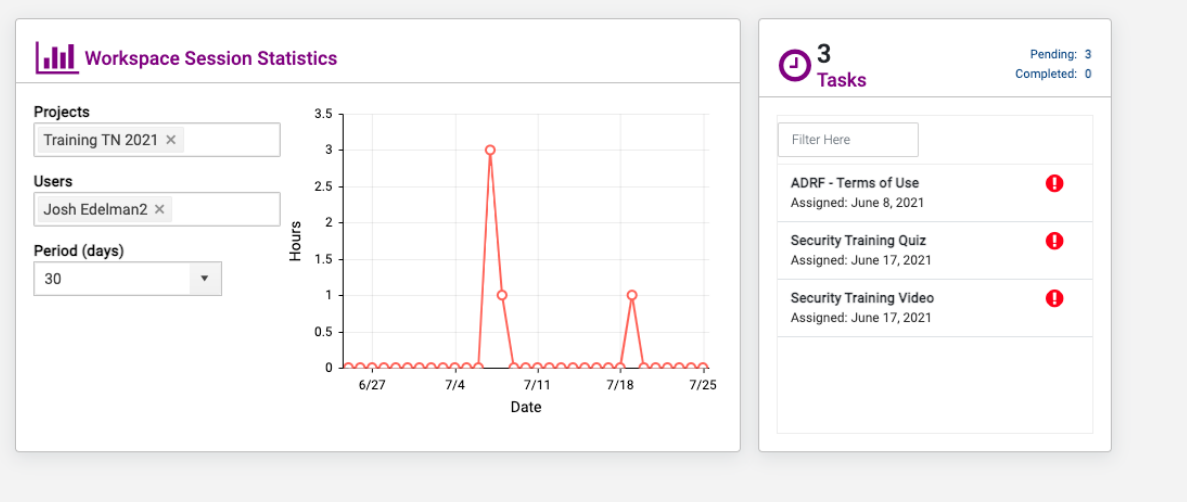
Thee Data Stewardship web-based application is positioned primarily as the management and monitoring console for project and data stewards. It provides detailed insight on project configurations, user activity, user onboarding status, and overall cost of a project on the ADRF. We focus on four primary pillars of information a Project/Data Steward most often focuses on:

- People – Who are the members of projects, how often do they use the ADRF, what exports have they requested and their status, estimated cost per person/project for current month and for the project since inception, and detailed usage metrics.
- Projects – Details of project start/end dates, abstract description, number of members onboarded and pending, and resources the project has access to (i.e. datasets, etc).
- Datasets – Description of the dataset, location on the ADRF (database or file system), size, name of the data steward(s), and the link to Enterprise Data Catalog (Informatica) describing the dataset and metadata.
- Agreements – What agreements are related to these projects, indication of each member's signing status, members pending signature, and term (dates) covered by the agreement(s).

As mentioned, the data stewardship application will track your ADRF usage. The app will also consolidate your ADRF Terms of Use, Security Training Quiz, and Security Training Video into one place. In order to complete ADRF onboarding, all three of the mentioned tasks are to be completed by the user (researcher). To access the Data Stewardship app, log in using your credentials at https://adrf.okta.com and click on the Data Stewardship icon. See picture below:

Once inside the Data Stewardship app, you have access to your personal workspace sessions statistics and the three tasks. See the example below:





Projects

Training TN 2021 ×

Users

Josh Edelman2 ×

Period (days)

30 ▾

3 Tasks

Pending: 3

Completed: 0

Filter Here

ADRF - Terms of Use
Assigned: June 8, 2021

Security Training Quiz
Assigned: June 17, 2021

Security Training Video
Assigned: June 17, 2021

## 2.2 ADRF - Terms of Use

The Terms of Use need to be completed before you are given access to the data and project sace inside the ADRF. To complete ADRF Terms of Use, open the Data Stewardship app and click on ADRF - Terms of Use. This will direct you to an Docusign site to complete the signing of the agreement.

## 2.3 Security Training Video

The Security Training Video needs to be completed as well. To complete the training, open the Data Stewardship app and click on Security Training Video. This will direct you to the video; click Mark Complete when you have completed this training.

## 2.4 Security Training Quiz

The Security Training Quiz needs to be completed after the Security Training Video. To complete the train- ing, open the Data Stewardship app and click on Security Training Quiz. This will direct you to the quiz, where you must answer five out of six questions correctly to pass.

# 3 ADRF - FAQs

## 3.1 FAQs

**How do I set up my Multifactor Authentication**

You should be prompted to set up multifactor authentication when you create your account, the options are SMS, voice call, email and the Okta verify application.

**Can I set up more than one form of Multifactor Authentication?**

This is recommended. If you lose access to one form of MFA, you would still be able to gain access to your account using an alternative. To do so, please log on to https://adrf.okta.com and select your name on the top right and click settings. Here you can modify or set up your SMS, voice call, email or Okta multifactor authentication.

**How can I reset my password?**

You can use the "Need help signing in?" option on the sign on page (https://adrf.okta.com) which will send a link to your email to reset your password. You may have to verify your identify by answering security questions which you set up when creating your account.

**What if I do not remember my security questions or if I get locked out?**

You would have to reach out to support at support@coleridgeinitiative.org to have your account unlocked and you would have to reset your security questions so that you can recover your account in the future.

**I can log into the ADRF but my desktop and DS application just show blank pages.**

Please ensure the connection to ADRF is not being blocked by your organizations VPN and/or firewall (try using a device not connected to your organization's network) and reach out to support@coleridgeinitiative.org.

### I saved a file in the C: drive or in the Desktop. When I logged back in, the file is no longer there. Can you restore it?

The ADRF is a temporary workspace environment, files left on the desktop will be removed when you log out of your session, and we cannot restore these files. See section 5.2.1 Best practice is to store files in your user folder on the U: drive

### How do I open an ipynb notebook?

On the desktop you should find an icon for JupyterLab, when you click that, a command prompt and a browser window are opened up, leave the command prompt running. You should be able to open the file by selecting File -> Open From Path  and providing the path to the folder containing the ipynb notebook.

### How can I ingest publicly available data into the ADRF?

Please open a support request by sending an email to support@coleridgeinitiative.org. Include the dataset you wish to have available inside the ADRF and documentation that confirms that the dataset is public.

### Where can I access publicly available data from within the ADRF?

Publicly available data is stored in the schema ds_public_1.

### Where is my project or training related data stored?

All project and training related databases are prefixed with 'pr_' (for project) or 'tr_' (for training). You may use this space when creating intermediate datasets or as a "working space". All project members have read and write access to this area (specific to your project).

### My data is not in a relational format. Where can I find these files?

Read-only non-relational data are stored in the G:\ drive on Windows Explorer. Project specific non-relational data and files are stored in project specific folders that are prefixed with 'pr_' or 'tr_'. The location of these folders are in the P:\ drive on Windows Explorer.

### What is the difference between the P:, U: and G: drives?

Each drive location has a different purpose and access rule:

P: Project specific files shared by ALL project members

U: User personal space. Only the user has read/write access to this area.
G: Non-relational datasets. Read-only access to authorized users only.

**I need to process a large amount of relational data. What is the destination location?**

The best practice is to process the data where it is currently located. If the data is in a relational database, perform as much of your processing using Redshift to make the most efficient use of resources (i.e. filtering, sorting, etc).

# 4 Do's and Don'ts For Discussing Data Inside the ADRF

It is important to protect the confidential data that is inside the ADRF in communicating with your team-mates. The general rule is that you should never take any exact number out of the ADRF. This means you should never write down or share any number by text, screenshot, or share an image even with a team-mate. The rules have become more complicated now that everything is online, because even though your team-mates are “safe people”, and zoom conversations are password protected and encrypted, we'd rather err on the side of caution when sharing information over zoom.

This cheat sheet summarizes some of the rules that apply to discussing data **before it has been exported from the ADRF and passed the ADRF team's disclosure review.** If you are unsure about a specific situation, please ask a Coleridge at support@coleridgeinitiative.org.

## 4.1 Exact Numbers

Do not describe a statistic in exact numbers. If you would like to communicate these values while not in person, you can have a private discussion via the projects drive inside the ADRF.

*Example*: If an average within a specific group was 5,000, you would need to convey this average on the projects drive.

## 4.2 Comparing Values

When comparing values, you are permitted to say that one value is more than, less than, or about the same as another. However, you cannot refer to the exact difference between the two numbers.

In practice, you can use pluses and minuses to convey differences between values for data that has not been exported from the ADRF.

*Example*: “The mean for Group A was roughly the same as the mean for Group B, but these values were both greater than that of Group C.”

## 4.3 Percentages/Proportions

Percentages and proportions also cannot be directly mentioned. Instead, you can refer to the percentage/proportion within 25%.

*Example*: If a proportion was 30%, you could say "The proportion is about 25%" or "The proportion is between 25% and 50%."

# 5 Accessing and Using Your Workspace

## 5.1 Logging into and Logging out of the ADRF

This video linked below runs through the necessary steps for logging into and logging out of the ADRF. If the video does not play, click here.

https://www.youtube.com/watch?v=_-AE_iOyF9w

## 5.2 Virtual Desktop Environment

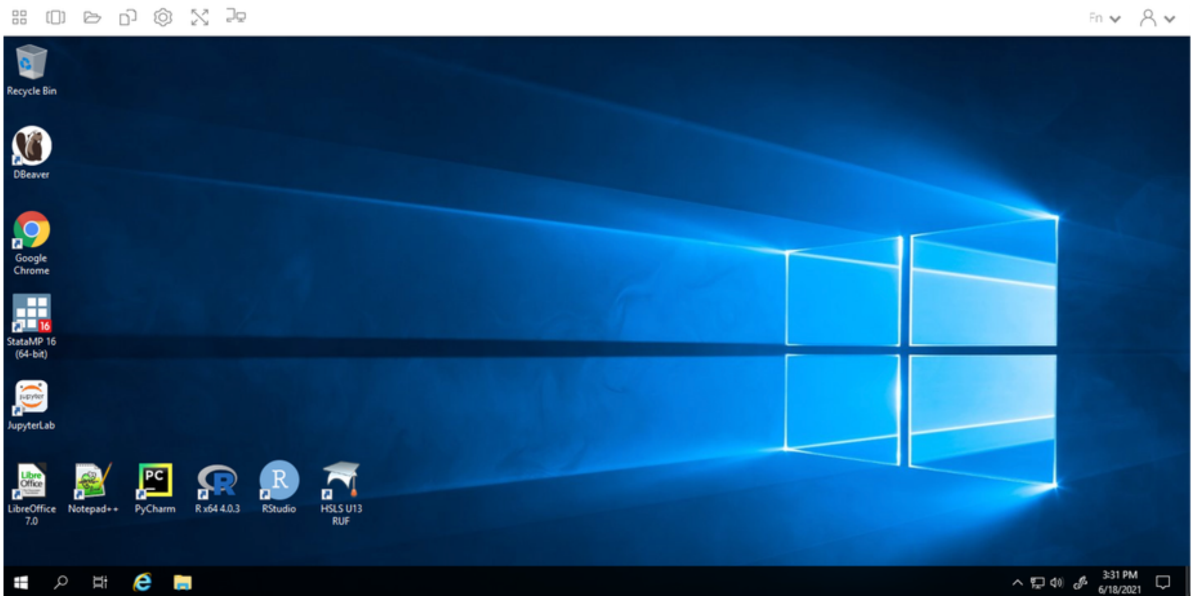
### 5.2.1 What is a VDE?

*Purpose, Contents, Capabilities*
A virtual desktop environment (VDE) allows you to interact with a remote system as if it were your own personal computer. The majority of your standard desktop functions are available, but the programs, data, and permissions are all controlled by the remote administrator (Coleridge Initiative). Thus, you will be working in a familiar environment while accessing protected data, programs, and systems that would otherwise be difficult to distribute. The ADRF uses a standard Windows environment (Windows Server) and provides a variety of software packages to conduct your analysis. For more on Windows capabilities, see the section on Windows Settings.

### 5.2.2 Temporary Nature of the Environment

While the environment is similar to that on your home computer (for Windows users), there are a handful of key differences. The first is that the environment is temporary in nature. This means that if you are not using it for a prolonged period of time (default is four hours but can vary by project), running programs will stop running and the information stored in temporary locations will be deleted. You will receive on on-screen message before any sessions are terminated. For more on safe, non-temporary storage locations in the ADRF, see the section on Storing Analytic Results.

Given the temporary nature of the ADRF, it is crucial to make sure that your work is saved—and saved in an appropriate location. Once this is complete and you are finished working,



make sure that you log out of the ADRF instead of closing the window. To do this, click the rightmost icon on the top taskbar to open up the dropdown menu and select End Session. You will be prompted to double-check that your work is saved prior to ending your session and confirm that you want to end your session.

## 5.3 Modifying the Environment

### 5.3.1 Establishing Personal Folders

Establishing your own personal folders is one of the simplest, yet most important, steps to take when setting up your environment. As we note in the section on Storing Analytic Results, the two possible places to store your analytic results or files are in either the U: drive or the P: drive.

You will find your personal folder in the U: drive. The folder name will include your Firstname and Lastname, and may additionally include your project workspace number. This is a personal workspace that only you can access in the ADRF.

### 5.3.2 The U: Drive and the P: Drive

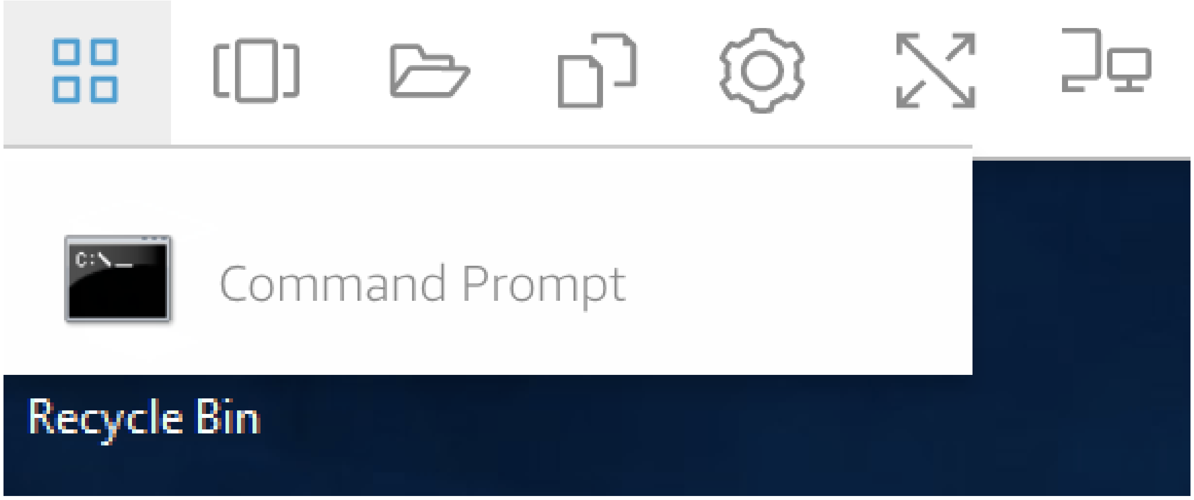
The U: drive is your user drive; it's where you will store any files you are working on. Only the user will have access to the U: drive. For example, if user A wants to share information with user B who is on the same project, user A will need to save files to a P: drive folder and not folders in their U: drive since user B will not be able to access user A's U: drive.

The P: drive is the project drive, which will be used to house project-specific folders. Thus, you and other collaborators on the same project will be able to save files to project drive folders.

Both the U: drive and P: drive have defined resource limitations of 150GB. When the workspace exceeds these limits, users will not be able to create new files or save data. The ADRF will not alert users when they approach on 150GB used. Users can check their current usage by right clicking on the user folder and clicking on properties.

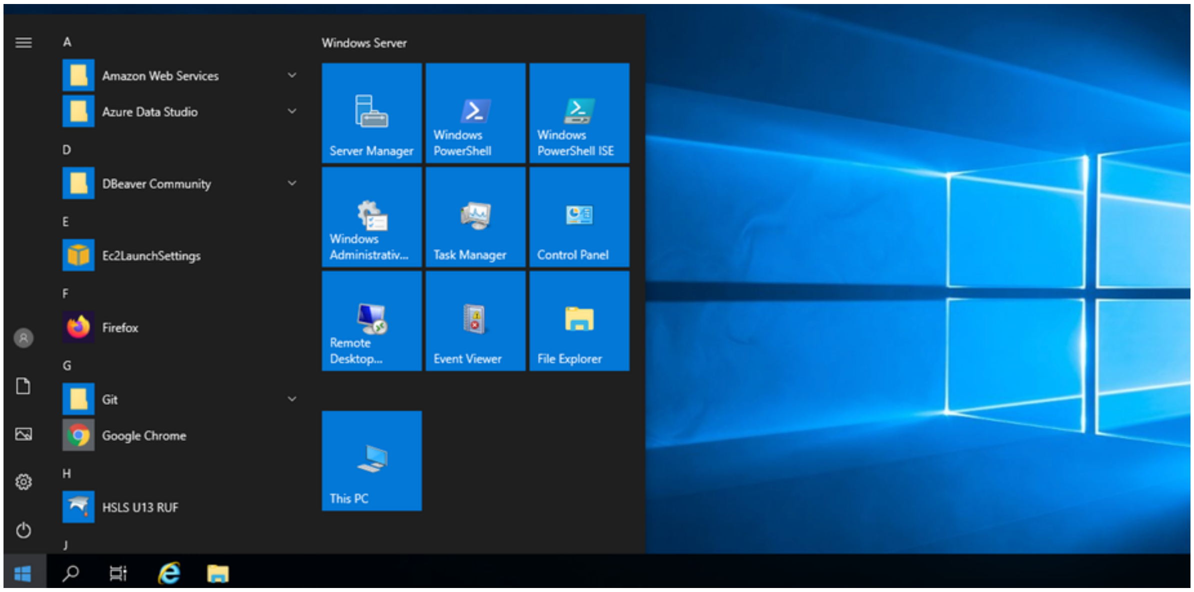
### 5.3.3 Other Modifications

The top taskbar contains shortcuts to the command prompt, multiple desktop windows, a temporary folder, settings, full-screen view, and toggling multiple monitors.



### 5.3.4 Windows Settings

Your desktop will look familiar if you are a Windows user. You will have icons for quick access to programs or browsers on your desktop. The windows icon on the bottom left side of the screen will open up a menu of programs, folders, and other tools, much as you would see on your own desktop. You will have access to PowerShell and several customization settings (e.g., remove bottom taskbar).



## 5.4 Software in the ADRF

### 5.4.1 JupyterLab

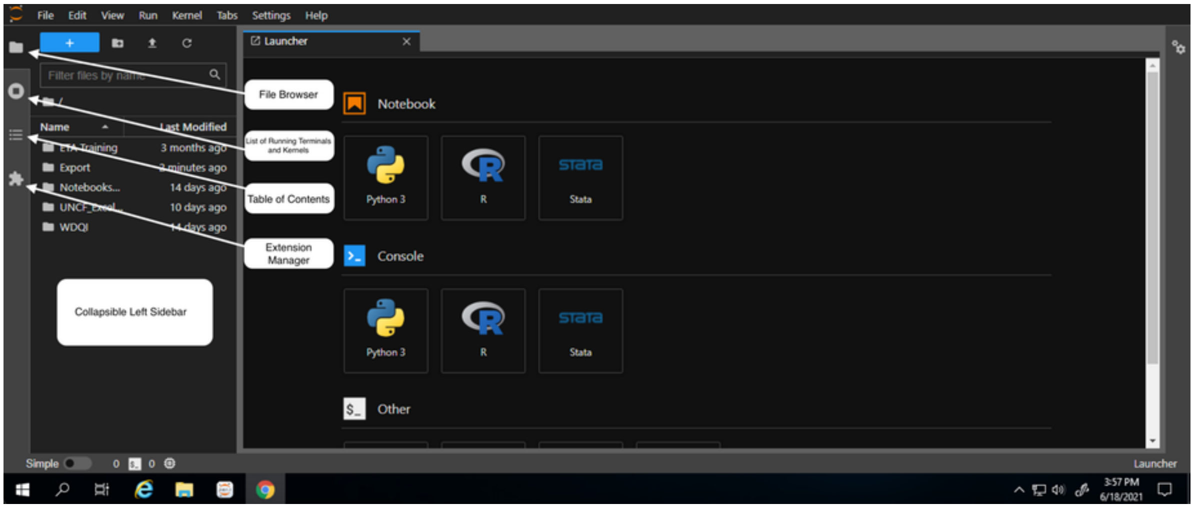
JupyterLab provides flexible building blocks for interactive, exploratory computing. While JupyterLab has many features found in traditional integrated development environments (IDEs), it remains focused on interactive, exploratory computing. For more on JupyterLab, see the interface documentation.

The JupyterLab interface on the ADRF consists of a main work area containing tabs of documents and activities, a collapsible left sidebar, and a menu bar. The left sidebar contains a file browser, the list of running terminals and kernels, the table of contents, and the extension manager.

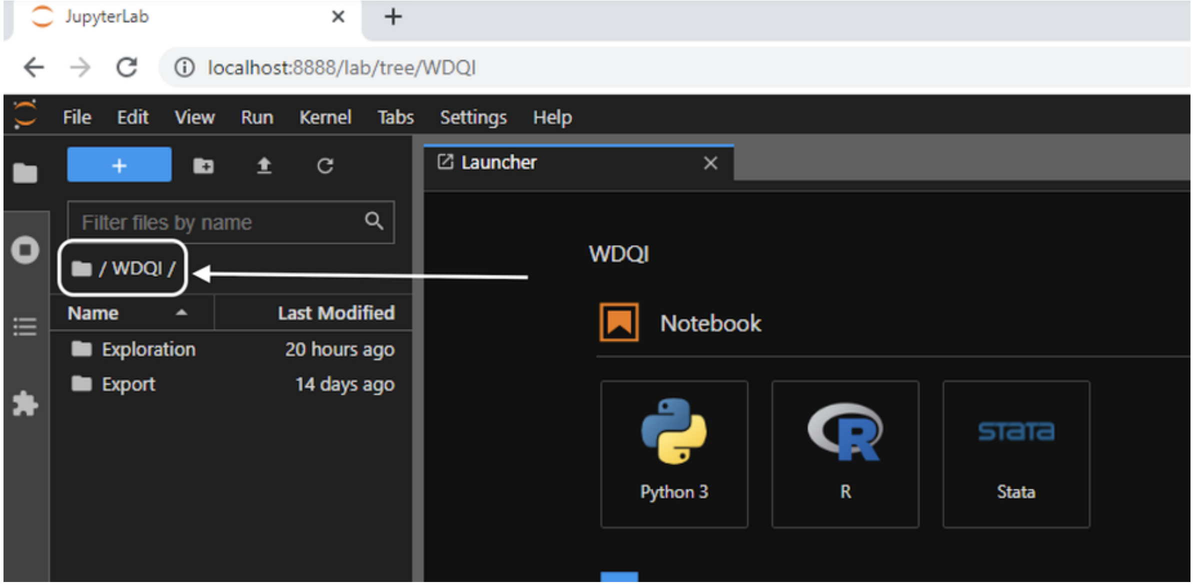
When using Jupyter Notebooks, make sure that all your work is saved to your U: drive and the correct director within the U: drive. You can find the active directory by reading the path displayed in the file browser. By default, JupyterLab opens with your U: drive as the base directory. Below, the folder icon in the white box is my user folder (not displayed, but titled `Firstname.Lastname`; you will have already set up your folder) and subfolder WDQI.

### 5.4.2 Notebooks

Jupyter Notebooks are documents that combine live runnable code with narrative text (Markdown), equations (LaTeX), images, interactive visualizations, and other rich output. You can



| Name | Last Modified |
|---|---|
| ETA Training | 3 months ago |
| Export | 2 minutes ago |
| Notebooks... | 14 days ago |
| UNCF_Excel... | 10 days ago |
| WDQI | 14 days ago |



localhost:8888/lab/tree/WDQI

| Name | Last Modified |
|---|---|
| Exploration | 20 hours ago |
| Export | 14 days ago |

create a notebook by clicking the blue + button in the file browser and then selecting a kernel (R, Python3, Stata) in the Launcher tab. For more information on getting started with Jupyter Notebooks, see JupyterLab Notebook documentation.

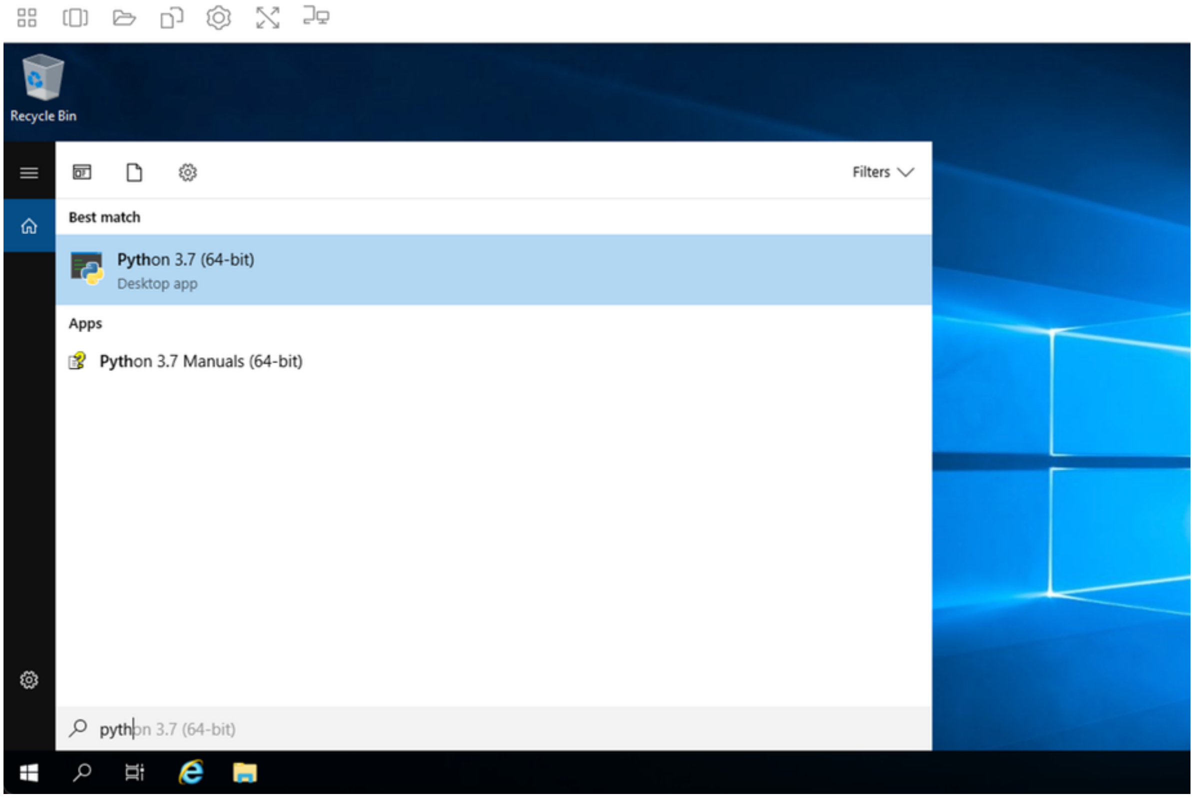
### 5.4.3 Accessing Stored Data from a Notebook

A common question is how to access stored data while writing to and using a Jupyter Notebook. Data in the ADRF are stored in a database using Microsoft SQL Server. For more information on how to access stored data in the ADRF based on choice of program (Python, R, Stata), see the section on Accessing Your Data.

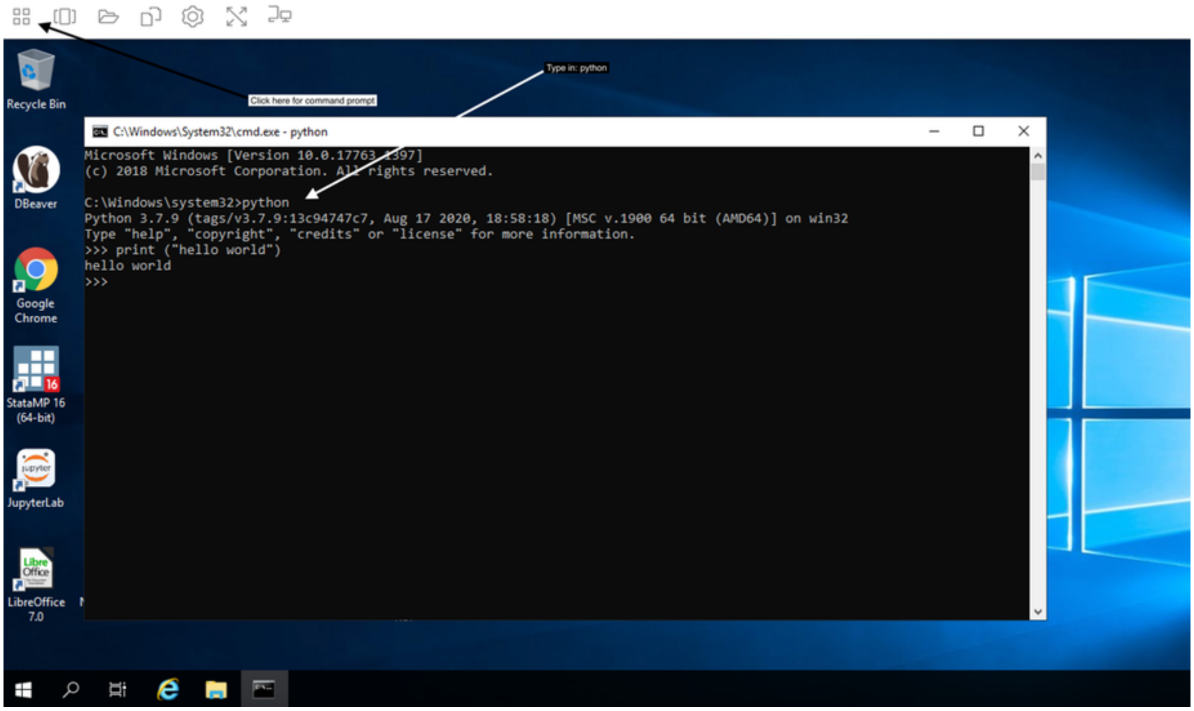
### 5.4.4 Python 3

Python is a general-purpose programming language. You can access Python in a multitude of ways:

1. Through the start menu (windows icon). Type in Python. A desktop app called Python 3.7 (64-bit) will populate a window where you can begin programming.



2. Through the command prompt in the top taskbar. Once the command prompt window is open, type in python.



```
Microsoft Windows [Version 10.0.17763.1397]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Windows\system32>python
Python 3.7.9 (tags/v3.7.9:13c94747c7, Aug 17 2020, 18:58:18) [MSC v.1900 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> print ("hello world")
hello world
>>>
```

3. Through JupyterLab. !is is the recommended way to access Python since it has packages installed and available, and an execution environment for testing and running code (as well as a place to write and save code). Open JupyterLab and make sure your directory is set appropriately in the “le browser. Once there, in your new Launcher window, click the Python 3 icon.

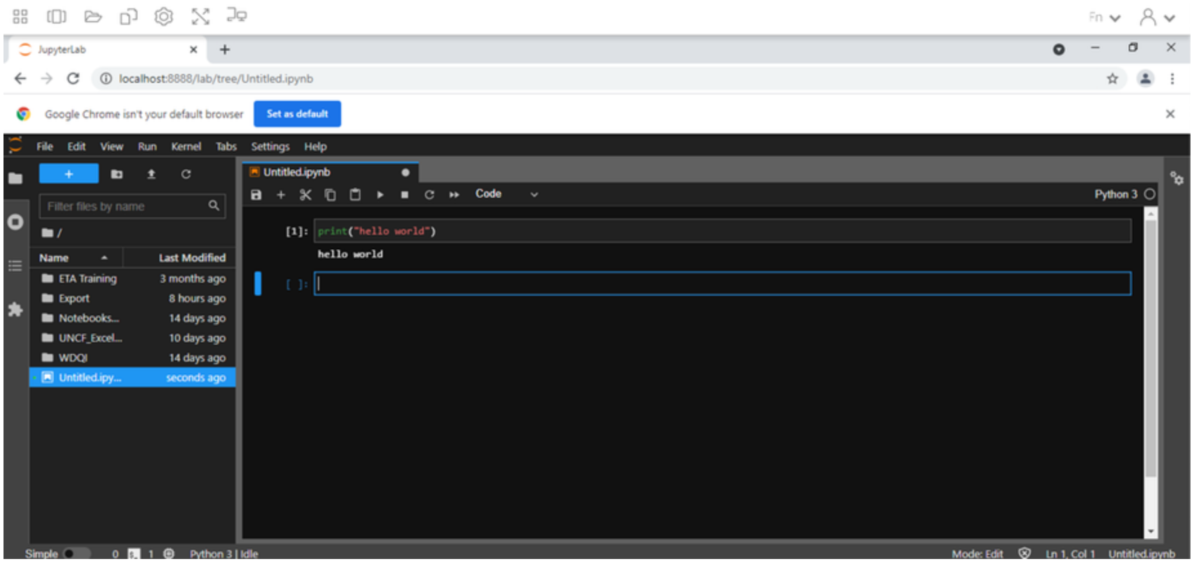
4. Through Pycharm.

### 5.4.5 R

R is a general-purpose programming language. You may access R in one of three ways:

1. Through RStudio. This is an integrated development environment (IDE) for R. You can run R code, display variables, debug R code, do inline visualizations, and more. Open RStudio through the desktop shortcut, or type RStudio in the start menu.

2. Through JupyterLab. Open JupyterLab and make sure your directory is set appropriately in the file browser. Once there, in your new Launcher window, click the R icon.



| Name | Last Modified |
|---|---|
| ETA Training | 3 months ago |
| Export | 8 hours ago |
| Notebooks... | 14 days ago |
| UNCF_Excel... | 10 days ago |
| WDQI | 14 days ago |
| Untitled.ipy... | seconds ago |

```
[1]: print("hello world")
     hello world
[ ]:
```

3. Through the R GUI (graphical user interface). Type R in the search bar and click to open the RGui.

### 5.4.6 Stata

Stata is a general-purpose statistical so#ware package. Stata can be accessed through the desktop shortcut StataMP 16 or by searching for it using the search or menu bar, or through JupyterLab.

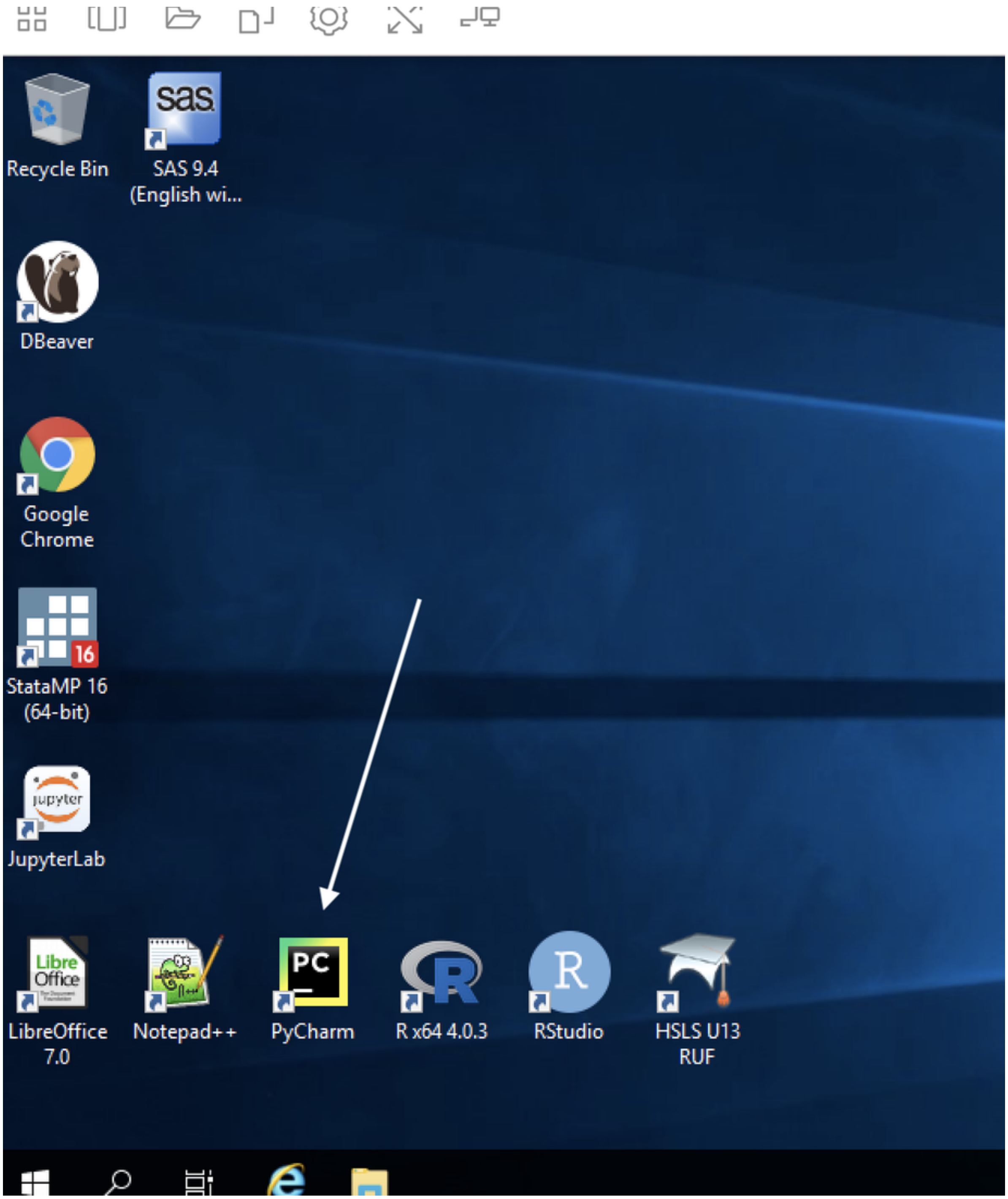
### 5.4.7 DBeaver

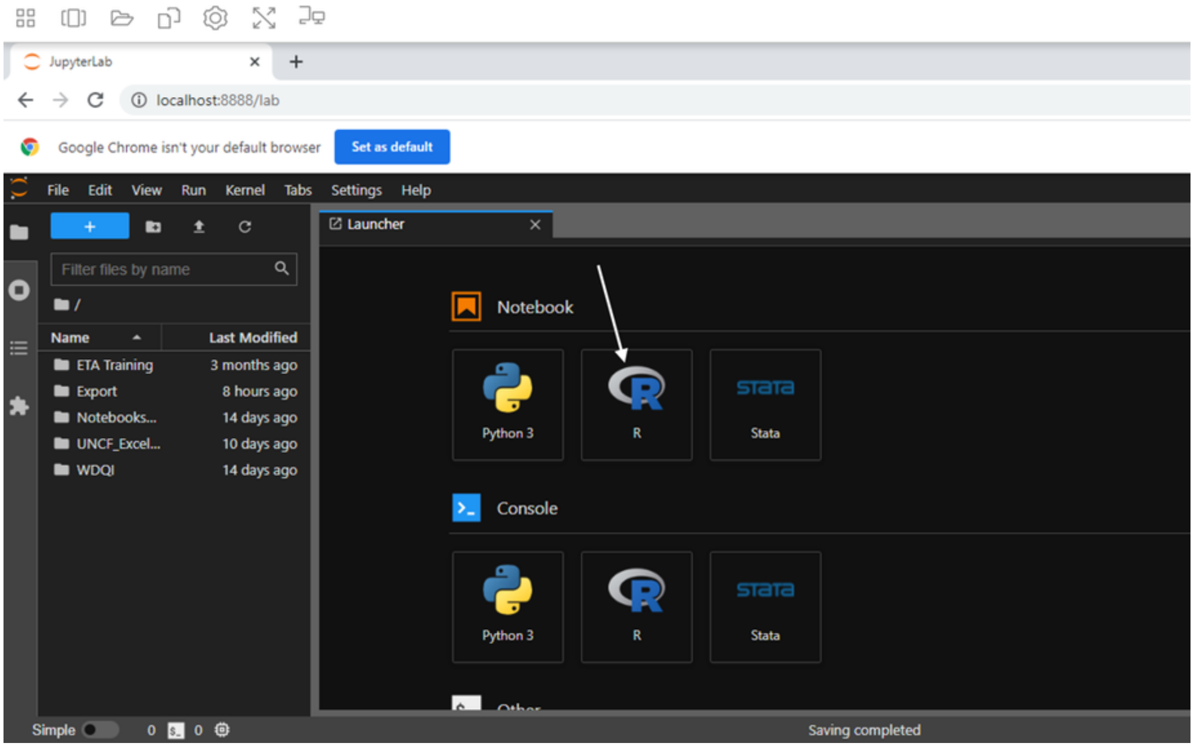
DBeaver is a universal tool for querying, editing, and managing data stored in Redshift databases. The ADRF stores data using AWS Redshift Server. DBeaver can be accessed through the desktop shortcut DBeaver or by looking it up using the search bar.

Once open, you will need to connect to a Redshift server. Please follow the directions in the Redshift Querying Guide Appendix 12.1 section of this guide to connect to the appropriate server.

### 5.4.8 Database Navigator

On the left side of DBeaver, a pane labeled Database Navigator allows you to easily explore what is in the server to which you are connected. By clicking the arrow, all the items within each server, Database, Schema, etc., are shown. When exploring these data and writing SQL


Recycle Bin
SAS 9.4 (English wi...
DBeaver
Google Chrome
StataMP 16 (64-bit)
JupyterLab
LibreOffice 7.0
Notepad++
PyCharm
R x64 4.0.3
RStudio
HSLS U13 RUF



| Name | Last Modified |
|---|---|
| ETA Training | 3 months ago |
| Export | 8 hours ago |
| Notebooks... | 14 days ago |
| UNCF_Excel... | 10 days ago |
| WDQI | 14 days ago |

queries, it is frequently useful to have the navigator expanded to see more easily what columns are in each table and their data type; the datatype can be seen in the screenshot to the right in parentheses next to each column name (e.g., clientid(char64) is a text column of length 64—for our purposes you can ignore the char... and varchar... and simply treat it as text).

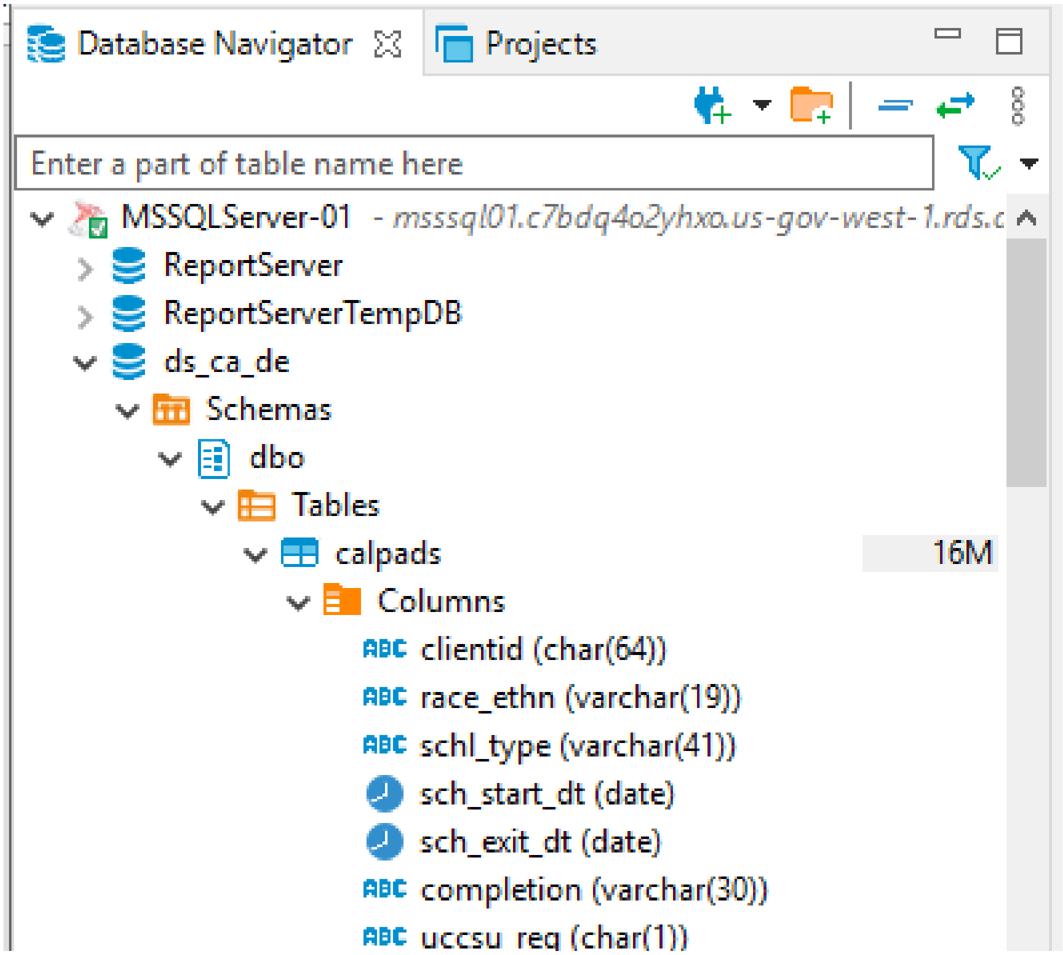
### 5.4.9 SQL Editor

The SQL Editor is where you can write your own queries to analyze the data. A new script can be opened by clicking on the blue almost-square (looks a bit like an unrolled scroll) on DBeaver's tool bar:

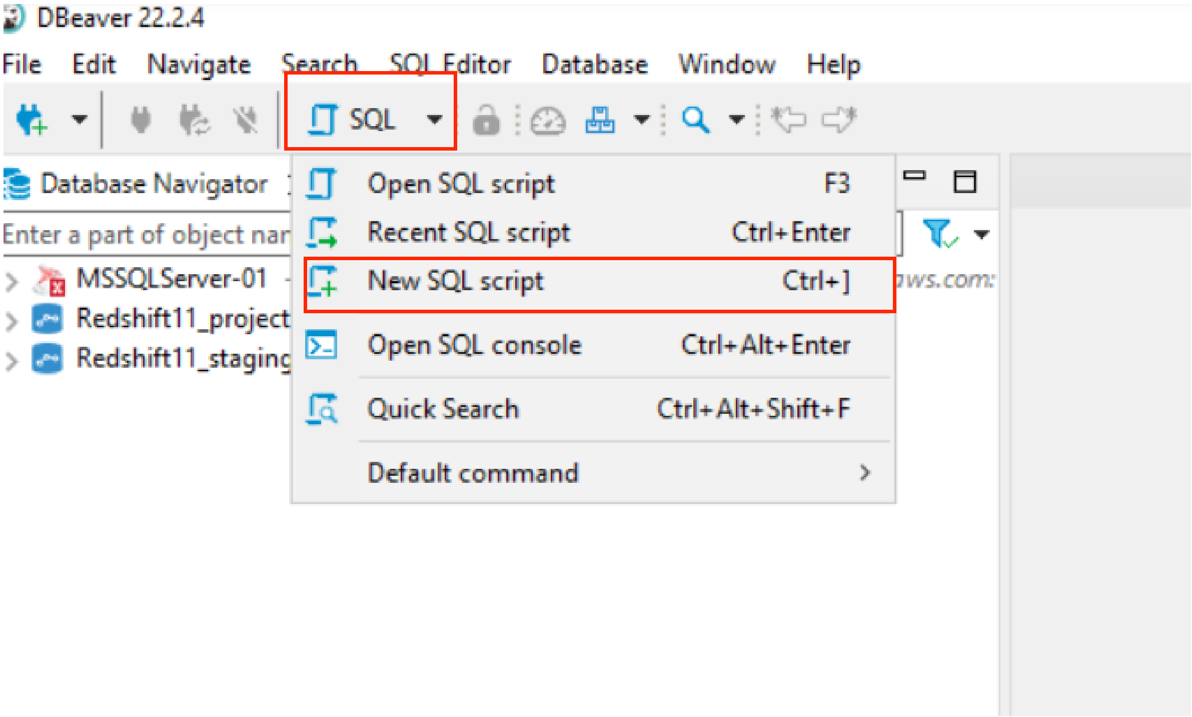
The location of this script button is circled in the red in the upper left of the screenshot below.

> Note: If you use the SQL button to open a new window, it will prompt you to select a data source and enter your username and password.

Once you have a SQL Editor window open, you can write a query and run it. One option to run a query is to use the keyboard to hit ctrl+enter, and another option is to use the orange triangle.



```
MSSQLServer-01  - msssql01.c7bdq4o2yhxo.us-gov-west-1.rds.c
  ReportServer
  ReportServerTempDB
  ds_ca_de
    Schemas
      dbo
        Tables
          calpads                                          16M
            Columns
              clientid (char(64))
              race_ethn (varchar(19))
              schl_type (varchar(41))
              sch_start_dt (date)
              sch_exit_dt (date)
              completion (varchar(30))
              uccsu_req (char(1))
```



### 5.4.10 Open a saved .sql FIle

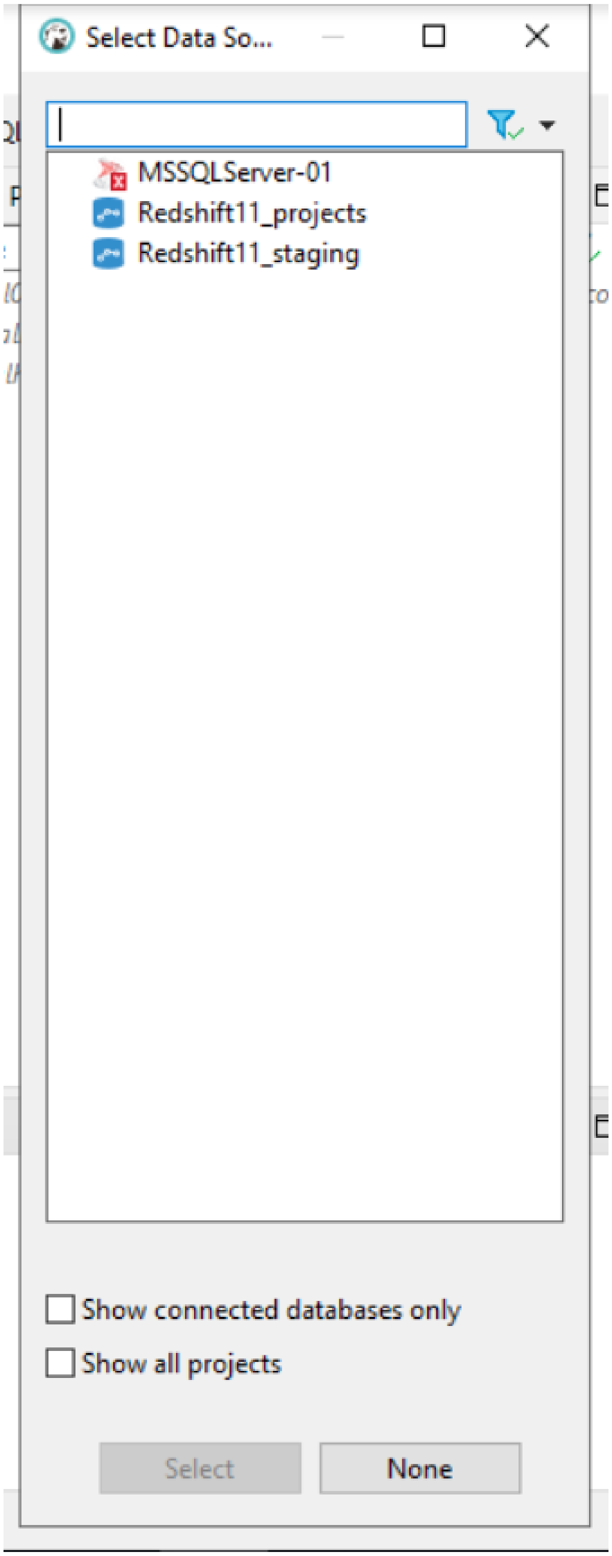
You do not have to create a new script every time! You can open a .sql file either by simply dragging and dropping it from the file explorer, or by going to File → Open File and navigating to a .sql file, as shown in the screenshot below:

Once you have done so, the top of your SQL Editor window should name the server connection inside the angle brackets to the left of the filename (<Redshift11_projects>).

### 5.4.11 LibreOffice

LibreOffice is an office productivity suite. LibreOffice comes equipped with six different programs: a word processor program (Writer), a spreadsheet program (Calc), a presentation program (Impress), a graphics editor program (Draw), a math equation program (Math), and a database management program akin to Microsoft Access (Base). LibreOffice may be accessed through the desktop shortcut DBeaver or by looking it up using the search bar. Once you've opened up LibreOffice, you can open any of those six programs, using the left sidebar. For more information on LibreOffice, visit the LibreOffice website.

Once you click on the icon, you'll see a page with a left sidebar that has a variety of document types under Create. Select the one suited to your needs and double click to open it.



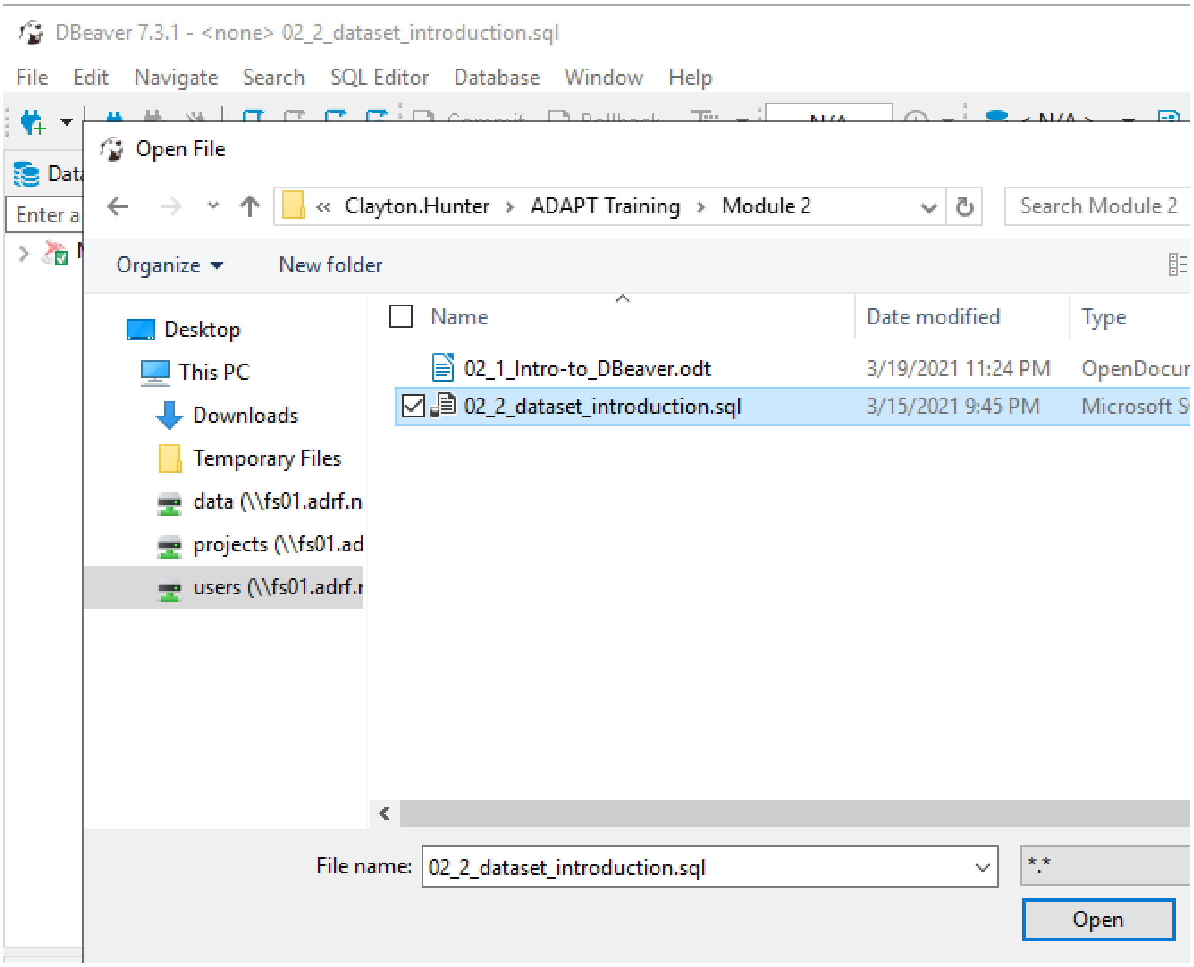
MSSQLServer-01

Redshift11_projects

Redshift11_staging

☐ Show connected databases only

☐ Show all projects

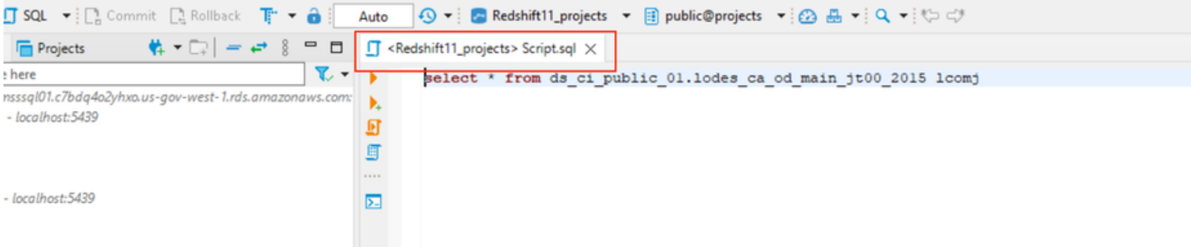


DBeaver 7.3.1 - <none> 02_2_dataset_introduction.sql

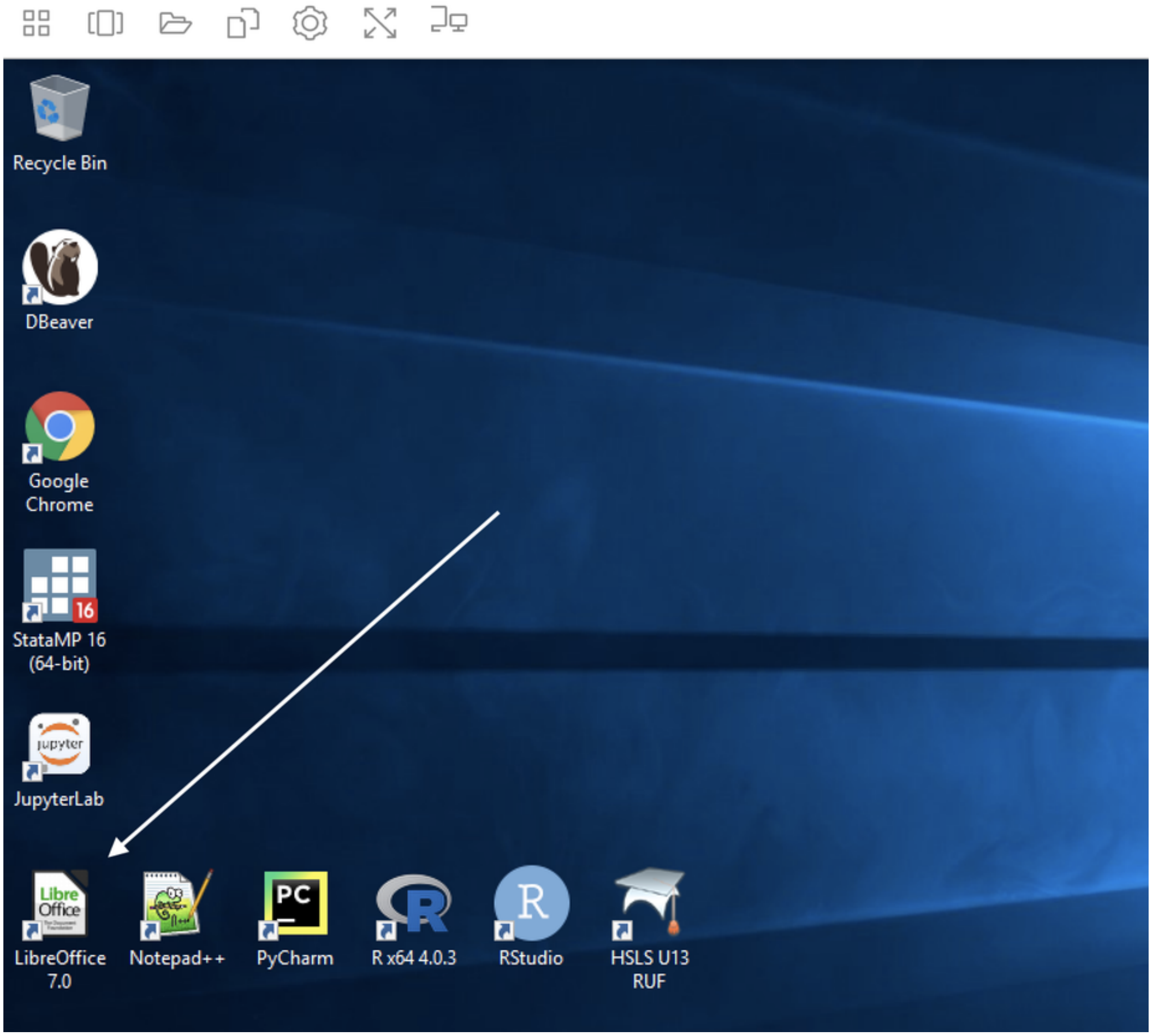
| □ Name | Date modified | Type |
|---|---|---|
| 02_1_Intro-to_DBeaver.odt | 3/19/2021 11:24 PM | OpenDocur |
| ☑ 02_2_dataset_introduction.sql | 3/15/2021 9:45 PM | Microsoft S |

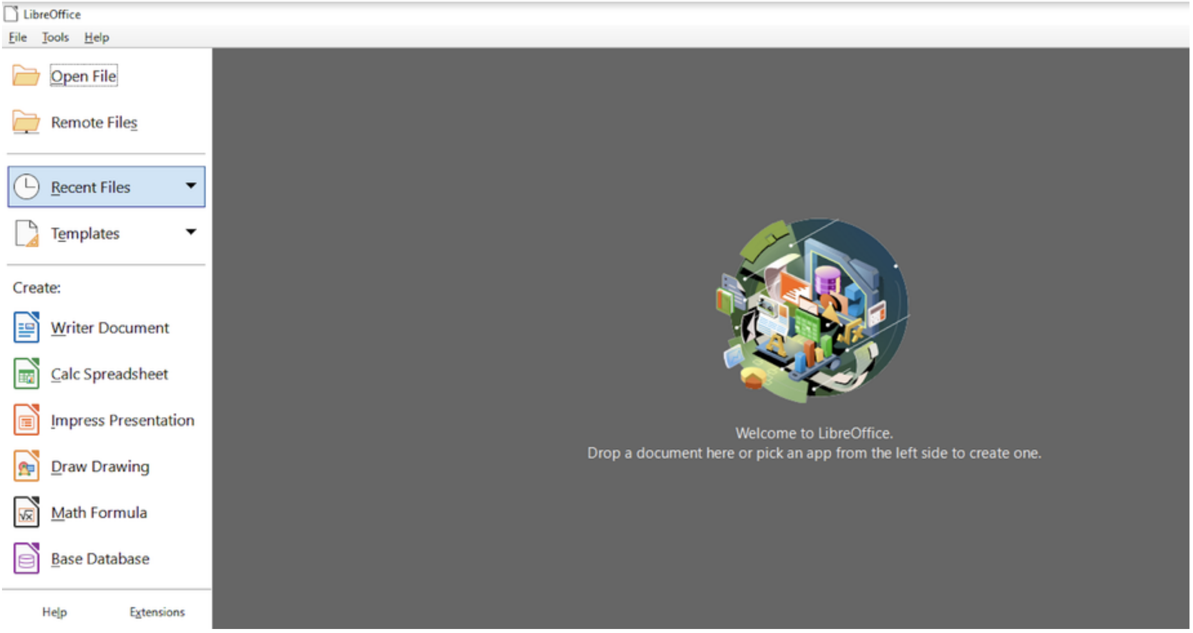
File name: 02_2_dataset_introduction.sql

*.*



```
select * from ds_ci_public_01.lodes_ca_od_main_jt00_2015 lcomj
```


Recycle Bin
DBeaver
Google Chrome
StataMP 16 (64-bit)
JupyterLab
LibreOffice 7.0
Notepad++
PyCharm
R x64 4.0.3
RStudio
HSLS U13 RUF



Welcome to LibreOffice.
Drop a document here or pick an app from the left side to create one.

### 5.4.12 More...

The ADRF provides a number of additional programs such as a simple text editor (Notepad++), PyCharm (an IDE for Python users), and several web browsers. Please note that web browsers are limited only to approved websites.

## 5.5 Available Software

The ADRF provides numerous software applications to users. Every user in the ADRF has access to:

- R Studio
- R
- Python, through Jupyter Labs or PyCharm
- Jupyter Labs, R and Python kernels available
- DBeaver
- LibreOffice
- Notepad++
- MikTex

- Java

If there is software that you would like to use for your project and is not installed in the ADRF, please email support@coleridgeinitiative.org.

If you would like to add additional packages to your workspace, please see the section on Adding Additional Packages in R/Python.

If there is a Python/R package you would like installed, please see Additional Packages in R/Python.

# 6 Accessing Your Data

## 6.1 Locate your Data in a Database

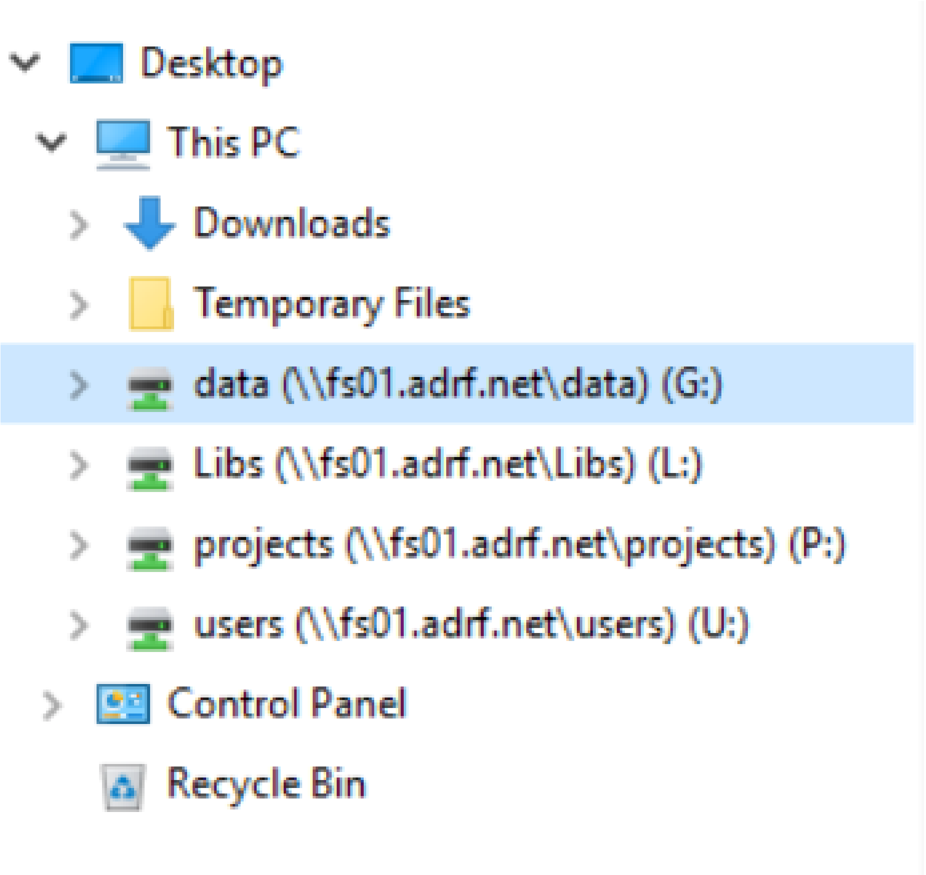
The ADRF stores data using Redshift. The simplest way to locate and get a quick overview of your data in a database is to use DBeaver. Please see the section on Data Organization, Amazon RedShift Querying Guide to locate and connect to your data.

### 6.1.1 G: Drive

Unstructured data is located on the `G:` drive inside the file system.

### 6.1.2 External Data and Code

Please note that importing of external data and code is restricted to only Coleridge staff. Given the secure and protected environment provided by the ADRF, all code, data, and packages that are coming from outside of the ADRF must be carefully vetted to prevent leaks, disclosure, or unauthorized access. This means that there is no direct method for uploading data or code from your system to the ADRF. Please contact support@coleridgeinitiative.org for any questions or assistance on importing your own code, data, or packages.



Desktop

This PC

Downloads

Temporary Files

data (\\fs01.adrf.net\data) (G:)

Libs (\\fs01.adrf.net\Libs) (L:)

projects (\\fs01.adrf.net\projects) (P:)

users (\\fs01.adrf.net\users) (U:)

Control Panel

Recycle Bin

# 7 Storing Analytic Results

## 7.1 Eligible Locations

### 7.1.1 User Drive

The U: drive is your user drive; it's where you will store any files you are working on. Only the user will have access to the U: drive. For example, if user A wants to share information with user B who is on the same project, user A will need to save files to a P: drive folder and not folders in their U: drive since user B will not be able to access user A's U: drive.

### 7.1.2 Project Drive

The P: drive also allows permanent storage. This drive is accessible by anyone on the same project, but not across projects. This is the only drive outside of the user drive where saved files will not be erased after logging out of the ADRF.

### 7.1.3 SQL

Each project will have a project-specific database created. All members of the project will have read and write permissions for data and may also create their own objects (tables, etc.). The project databases are prefixed with pr-.

## 7.2 Ineligible Locations

The G: drive (data), the L: drive (Libs), and the desktop are not eligible for long-term file storing. You won't have permissions to write to either the G: drive or the L: drive. The desktop will function only as temporary storage—as soon as a user is logged out of the ADRF, your desktop will be cleared. Additionally, since Wi-Fi connectivity can be imperfect, desktop storage for any amount of time is not recommended.

## 7.3 Storage Size Restrictions

Storage size varies by project, but is capped at a predetermined amount. Additional storage costs may vary depending on the resource requirements. https://aws.amazon.com/appstream2/pricing/

## 7.4 Best Practices

To save storage space, try not to save raw data tables—in particular, don't save copies of or large subsets of data that are already available through standard sources. Instead, access data through the methods described in the prior sections here, as appropriate for your programming language or program.

Organize folders in a way that makes sense for your particular project. For example, you might have folders for a particular analysis or sub-projects. Dates on file names can be helpful for version control.

Keep tabs on how much storage you are using compared to the allocated amount of storage.

# 8 Sharing Information within the ADRF

## 8.1 ADRF Messenger

The ADRF messenger is an internal collaboration tool and will be made available once testing is complete.

## 8.2 Shared Folders

Shared folders within a project are a great way to share information with other members on a team project. Remember that when working with teams you may not share the ADRF screen (even project folders) with other members on video platforms or otherwise, whether or not your team members are working on the same project.

## 8.3 Sharing Restrictions

Again, remember that when working with teams you may not share the ADRF screen with other members on video platforms or otherwise, whether or not your team members are working on the same project.

The information contained in the ADRF is restricted to reside only in the ADRF for all purposes unless it passes Export Review. This means that it cannot be shared or potentially shared with any unauthorized parties. Do not write down any numbers or figures or tables corresponding to data in the ADRF. Copying and pasting is restricted, but manually circumventing this is also not permitted by your data agreements.

# 9 Export guidelines

## 9.1 Export Review Guidelines

To provide ADRF users with the ability to draw from sensitive data, results that are exported from the ADRF must meet rigorous standards meant to protect privacy and confidentiality. To ensure that those standards are met, the ADRF Export Review team reviews each request to ensure that it follows formal guidelines that are set by the respective agency providing the data in partnership with the Coleridge Initiative. Prior to moving data into the ADRF from the agency, the Export Review team suggests default guidelines to implement, based on standard statistical approaches in the U.S. government [1, 2] as well as international standards [3, 4], and [5]. The Data Steward from the agency supplying the data works with the team to amend these default rules in line with the agency's requirements. If you are unsure about the review guidelines for the data you are using in the ADRF or if you have any questions relating to exports, please reach out to support@coleridgeinitiative.org before submitting an export request.

To learn more about limiting disclosure more generally, please refer to the textbook or view the videos.

### 9.1.1 General Best Practices for a Successful Export

1. Currently, the review process is highly manual: Reviewers will read your code and view your output files, which may be time-consuming.
2. Each additional release adds disclosure risk and therefore limits subsequent releases; we ask that users limit the number of files they request to export to just the outputs necessary to produce a particular report or paper. If you are requesting an export of more than 10 files, there may be an additional charge.
3. The reviewers may ask you to make changes to your code or output to meet the requirements of guidelines that have been given by the providers of the data in the ADRF. Thus, we strongly encourage you to produce all output files—tables with rounded numbers, graphs with titles, and so forth—through code, rather than manually.
4. We ask that you only request review of final versions of output files, rather than in-progress versions. Any file containing intermediate output will be rejected.

5. Every code file should have a header describing the contents of the file, including a summary of the data manipulation that takes place in the file (e.g., regression, table or figure creation, etc.).
6. Documenting code by using comments throughout is helpful for disclosure reviews. The better the documentation, the faster the turnaround of export requests. If data files are aggregated, please provide documentation on the level of aggregation and for where in the code the aggregation takes place.
7. To help reviewers, who may not have seen your code before, we ask that users create meaningful variable names. For instance, if you are calculating outflows, it is better to name the variable "outflows" than to name it "var1."

### 9.1.2 Timelines for Export Process

1. Coleridge reviewers have five business days to complete an export from the day you submit an export request. However, timelines may differ depending on your agency, so please refer to your specific agency's guidelines.
2. The review process can be delayed if the reviewer needs additional information or if the reviewer needs you to make changes to your code or output to meet the ADRF nondisclosure requirements.

## 9.2 Preparing Data for Export

### 9.2.1 Tables

1. Cell Sizes

    1. Each agency has specific disclosure review guidelines, especially with respect to the minimum allowable cell sizes for tables. Refer to these guidelines when preparing export requests. If you are unsure of what guidelines are in place for the dataset with which you are working in the ADRF, please reach out to support@coleridgeinitiative.org.
    2. For individual-level data, please report the number of observations from each cell. For individual-level data, the default rule is to suppress cells with fewer than 10 observations, unless otherwise directed by the guidelines of the agency that provided the data.

3. If your table includes row or column totals or is dependent on a preceding or subsequent table, reviewers will need to take into account complementary disclosure risks—that is, whether the tables' totals, or the separate tables when read together, might disclose information about individuals in the data in a way that a single, simpler table would not. Reviewers will work with you by offering guidance on implementing any necessary complementary suppression techniques.

2. Weighted Data

    1. If weighted results are to be exported, you must report both weighted and unweighted counts.

3. Ratios

    1. If ratios are reported, please report the number of valid cases for both the numerator and the denominator (e.g., number of men in state X and number of women in state X, in addition to the ratio of women in state X).

4. Percentiles

    1. Do not report exact percentiles. Instead, for example, you may calculate a "fuzzy median," by averaging the true 45th and 55th percentiles.

5. Percentages

    1. For any reported percentages or proportions, the underlying counts of individuals contributing to the numerators and denominators must be provided for each statistic in the desired export.

6. Maxima and Minima

    1. Suppress maximum and minimum values in general.

    2. You may replace an exact maximum or minimum with a top-coded value.

### 9.2.2 Graphs

1. Graphs are representations of tables. Thus, for each graph (which may have, e.g., a jpg, pdf, png, or tif extension), provide the source data of the underlying table of the graph following the guidelines for tables above.

2. Because graphs and other figures take the most time to review, the number of generated graphs should be as low as possible. Please consider the possibility that you could export the underlying table instead, and generate the graph in another package.

3. If a graph is produced from aggregated data or from tables that have been disclosure-proofed following the guidelines above (e.g., bar charts of magnitudes), provide the underlying tables.
4. If a graph is produced directly from unit-record data but aggregated in the visualization (e.g., frequency histograms), provide the underlying tables.
5. If a graph is produced directly from unit-record data and displays unit-record values (e.g., scatterplots, plots of residuals), the graph can be released only after you ensure that individuals cannot be re-identified and that values can only be estimated with a high level of uncertainty. Further processing to meet this requirement can include, but is not restricted to, cutting off the tails of a distribution, removing outliers, jittering the actual values, and removing or modifying axis values.
6. If a graph is produced from the results of modeling or derivation and uses the unit-record data (e.g., regression curves), the graph can be released only if the values cannot be used to find original data values.

    1. Graphs of this type are generally automatically cleared.
    2. For precision/recall graphs, you will need to report the sample size used to generate your model(s).

### 9.2.3 Model Output

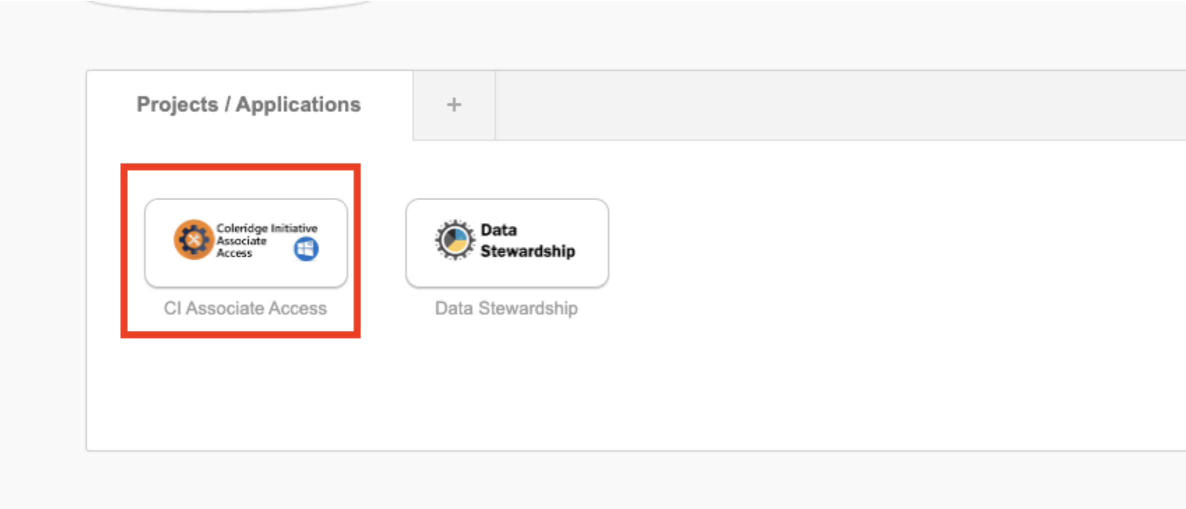
1. Output from regression or machine-learning models generally does not pose a risk of disclosing personally identifiable information, as long as the models are not based on small samples. Provide the counts for each variable that produces the model output. If categorical variables are used then provide the counts for each category.

## 9.3 Submitting an Export Request

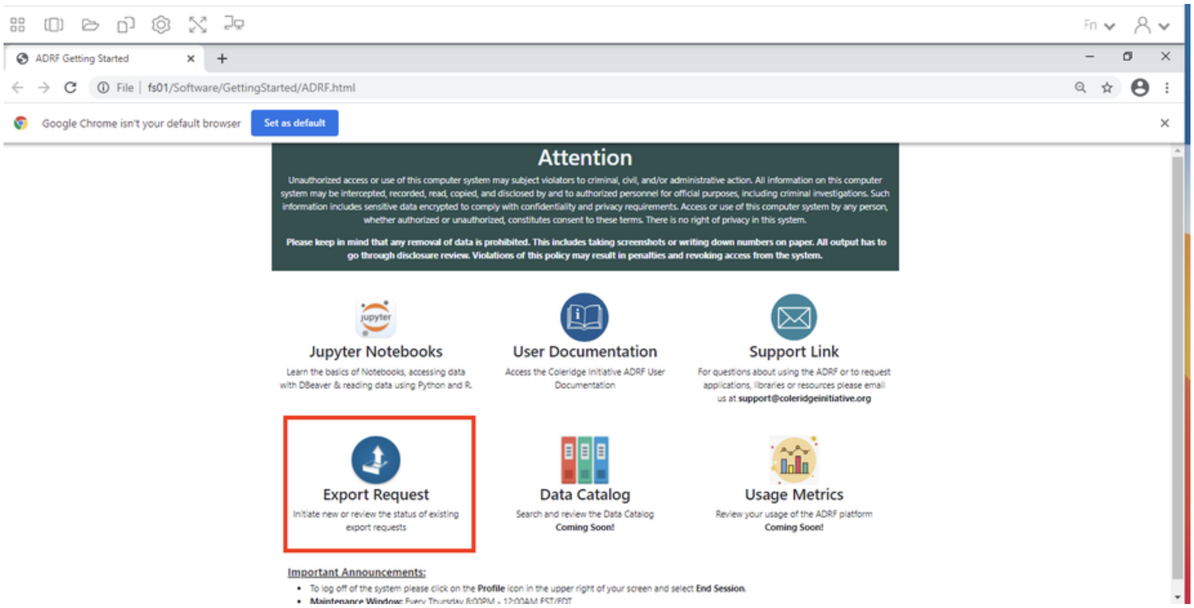
To request an export be reviewed, please watch the following video or follow the instructions below:

Export Video

1. Click here: http://adrf.okta.com (ADRF 3).
2. Input your login credentials.
3. Verify yourself with Okta (download Okta Verify on your smartphone or other device).
4. Choose your project as seen in the photo below. For the purpose of this document, you are seeing the Coleridge Initiative Associate Access project.



5. Select Desktop and login with the same credentials you had done previously.
6. Upon entering the ADRF, a chrome page will appear as shown in the photo below. On this page, click Export Request in the bottom left corner. Or, from the ADRF desktop, open Google Chrome and navigate to export.adrf.net. (Note: export.adrf.net is an address that will only work within the ADRF desktop).


ADRF Getting Started
File | fs01/Software/GettingStarted/ADRF.html
Google Chrome isn't your default browser
Set as default

## Attention

Unauthorized access or use of this computer system may subject violators to criminal, civil, and/or administrative action. All information on this computer system may be intercepted, recorded, read, copied, and disclosed by and to authorized personnel for official purposes, including criminal investigations. Such information includes sensitive data encrypted to comply with confidentiality and privacy requirements. Access or use of this computer system by any person, whether authorized or unauthorized, constitutes consent to these terms. There is no right of privacy in this system.

**Please keep in mind that any removal of data is prohibited. This includes taking screenshots or writing down numbers on paper. All output has to go through disclosure review. Violations of this policy may result in penalties and revoking access from the system.**

**Jupyter Notebooks**
Learn the basics of Notebooks, accessing data with DBeaver & reading data using Python and R.

**User Documentation**
Access the Coleridge Initiative ADRF User Documentation

**Support Link**
For questions about using the ADRF or to request applications, libraries or resources please email us at **support@coleridgeinitiative.org**

**Export Request**
Initiate new or review the status of existing export requests

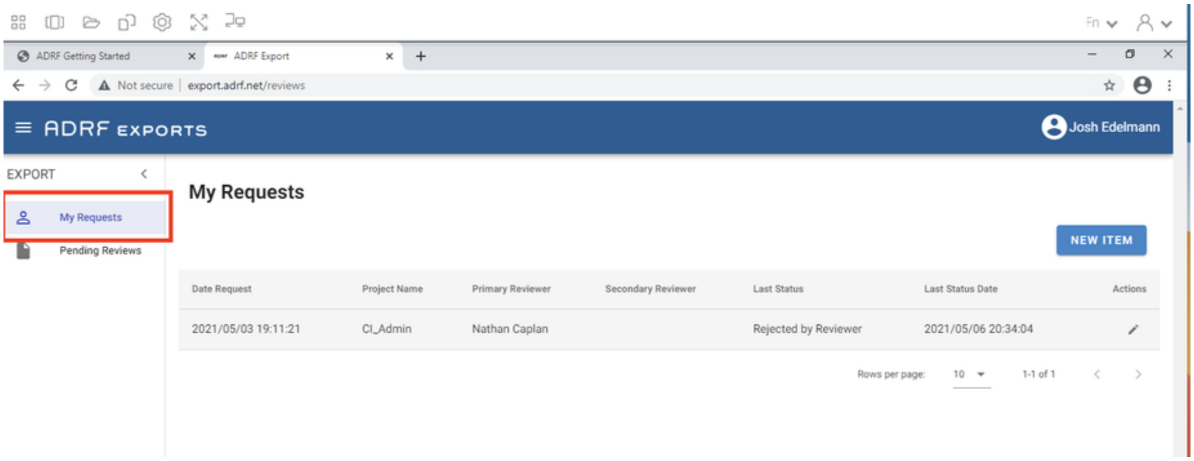
**Data Catalog**
Search and review the Data Catalog
**Coming Soon!**

**Usage Metrics**
Review your usage of the ADRF platform
**Coming Soon!**

Important Announcements:

- To log off of the system please click on the **Profile** icon in the upper right of your screen and select **End Session**.
- **Maintenance Window:** Every Thursday 8:00PM - 12:00AM EST/EDT

7. Click My Requests, or the top (person-shaped) icon, at the left side of the window as shown in the screenshot below.

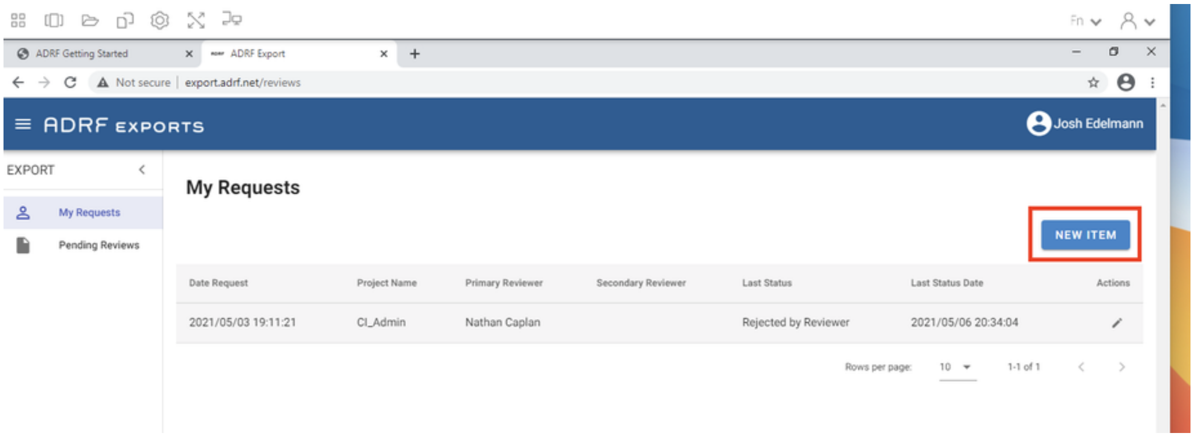


My Requests

| Date Request | Project Name | Primary Reviewer | Secondary Reviewer | Last Status | Last Status Date | Actions |
|---|---|---|---|---|---|---|
| 2021/05/03 19:11:21 | CI_Admin | Nathan Caplan | | Rejected by Reviewer | 2021/05/06 20:34:04 | |

Rows per page: 10 1-1 of 1

8. Click New Item as shown below



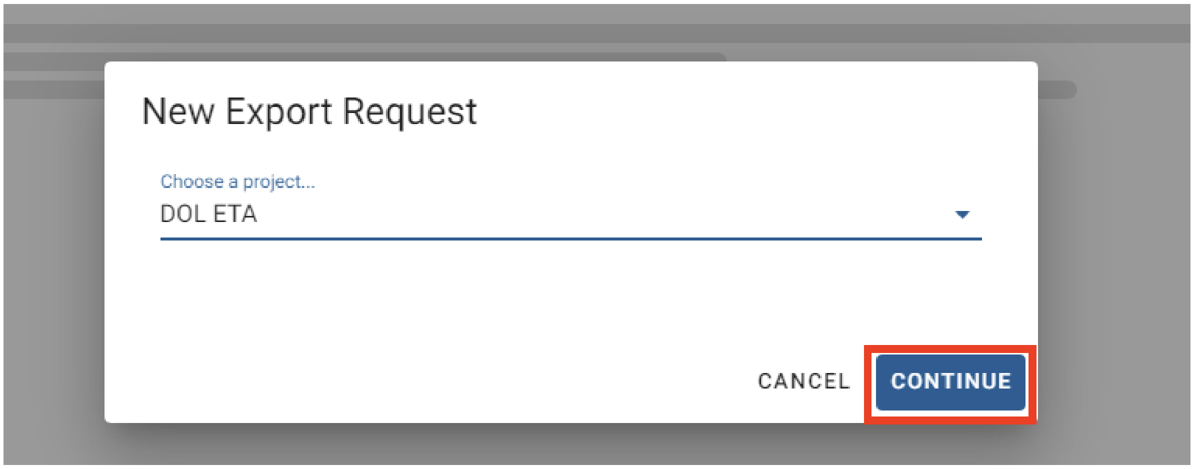
My Requests

| Date Request | Project Name | Primary Reviewer | Secondary Reviewer | Last Status | Last Status Date | Actions |
|---|---|---|---|---|---|---|
| 2021/05/03 19:11:21 | CI_Admin | Nathan Caplan | | Rejected by Reviewer | 2021/05/06 20:34:04 | |

Rows per page: 10 1-1 of 1

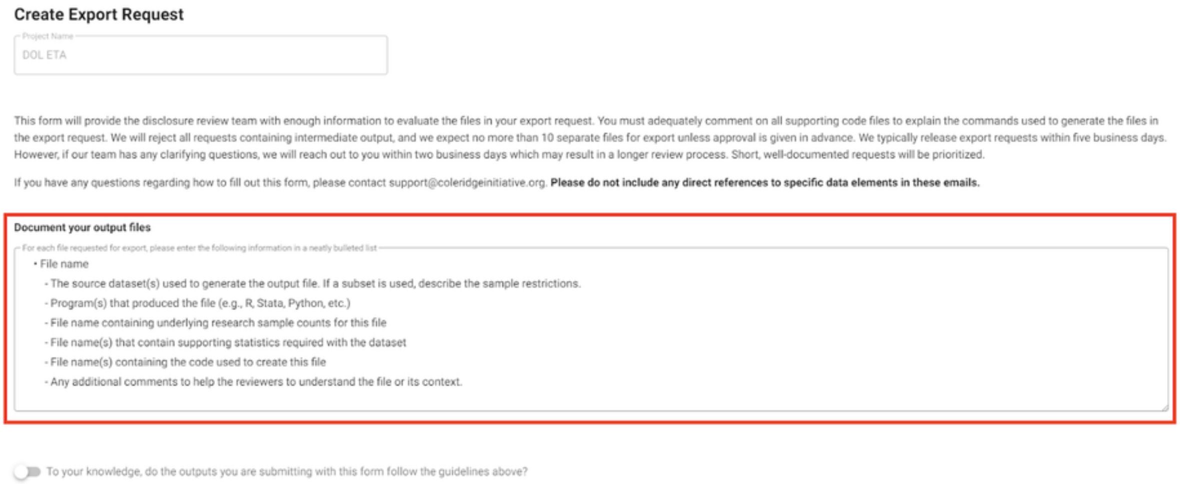
- You will be asked to select the project to which your export relates. If you do not see the correct project listed in the dropdown list, please reach out to our support team at support@coleridgeinitiative.org.
- After selecting a project, click Continue.

9. Read through the entire page that loads. This page, titled "Create Export Request," will ask for you to comment on all supporting code files to explain the commands used to generate the files in the export request. The Export Review team will reject all requests containing intermediate output, and there should be no more than 10 separate files for export unless approval is given in advance. The Export Review team will typically release export requests within five business days. However, if the team has any clarifying questions, this could result in a longer review process. You need to document your output files in the text box provided. See the example below:



New Export Request

Choose a project...
DOL ETA



Create Export Request

Project Name: DOL ETA

This form will provide the disclosure review team with enough information to evaluate the files in your export request. You must adequately comment on all supporting code files to explain the commands used to generate the files in the export request. We will reject all requests containing intermediate output, and we expect no more than 10 separate files for export unless approval is given in advance. We typically release export requests within five business days. However, if our team has any clarifying questions, we will reach out to you within two business days which may result in a longer review process. Short, well-documented requests will be prioritized.

If you have any questions regarding how to fill out this form, please contact support@coleridgeinitiative.org. **Please do not include any direct references to specific data elements in these emails.**

**Document your output files**

For each file requested for export, please enter the following information in a neatly bulleted list

- File name
  - The source dataset(s) used to generate the output file. If a subset is used, describe the sample restrictions.
  - Program(s) that produced the file (e.g., R, Stata, Python, etc.)
  - File name containing underlying research sample counts for this file
  - File name(s) that contain supporting statistics required with the dataset
  - File name(s) containing the code used to create this file
  - Any additional comments to help the reviewers to understand the file or its context.

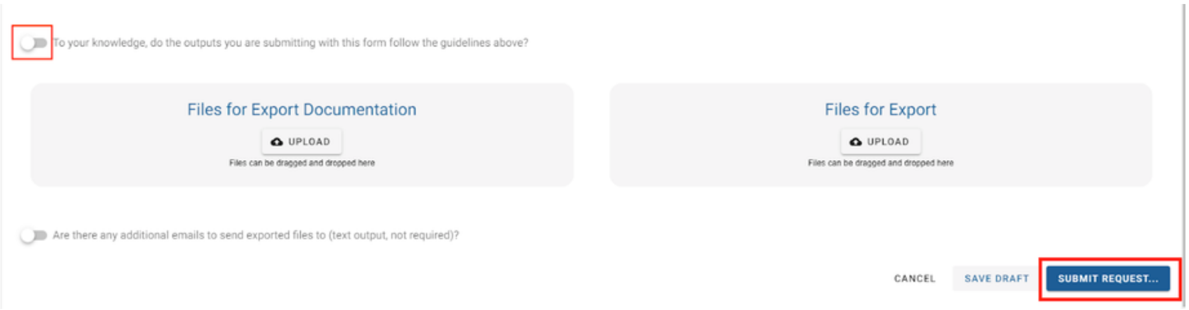
To your knowledge, do the outputs you are submitting with this form follow the guidelines above?

10. When you have read through and followed the page instructions, and are ready to proceed:

Move the slider at the bottom of the page to indicate that you have followed the page's guidelines.

At the bottom of the page, upload each of the files that you have prepared.

Click `Submit Request…` to create the export request.



To your knowledge, do the outputs you are submitting with this form follow the guidelines above?

Files for Export Documentation
UPLOAD
Files can be dragged and dropped here
Files for Export
UPLOAD
Files can be dragged and dropped here

Are there any additional emails to send exported files to (text output, not required)?

CANCEL
SAVE DRAFT
SUBMIT REQUEST...

11. You can click `My Requests` at the left side of the window to view your current and previous export requests.

    1. To learn more about exporting results, please watch these videos.

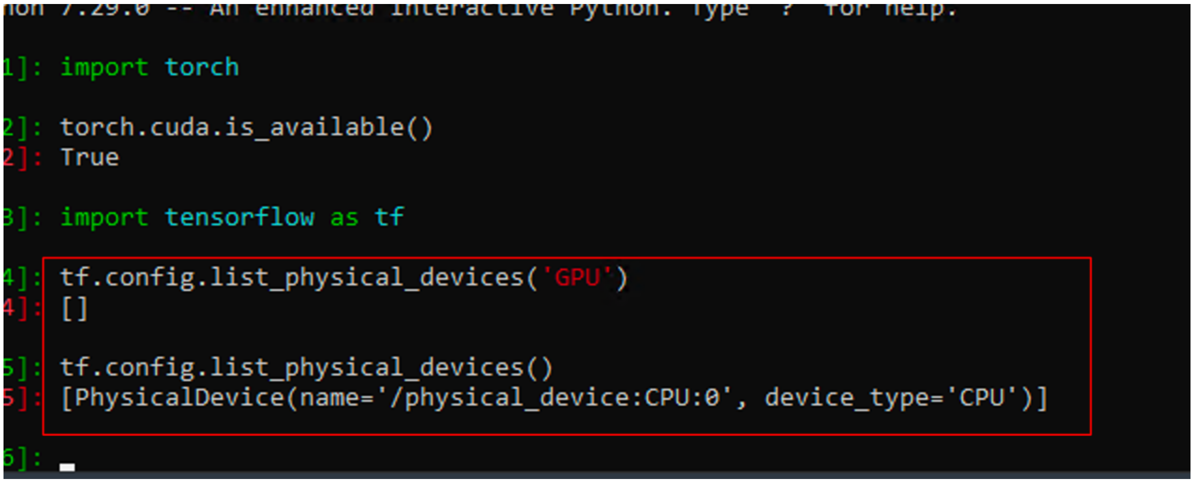
# 10 Adding Additional Packages in R/Python

## 10.1 Adding Additional Packages in R/Python

In order to request additional packages in R/Python, please email support@coleridgeinitiative.org with the following information:

1. Package name and configuration.
2. A list of dependencies (whether the packages depend on OS-specific libraries (DLLs, .so)).
3. A simple validation script to confirm the installation was successful.

The third item allows Coleridge to verify that the package is accessible from within your Python/R environment. The validation script below demonstrates importing a package (*tensorflow*) and running a functionality specific to that package (*config.list_physical_devices()*). If the install was unsuccessful, then the import or the function would fail to run.



```
hon 7.29.0 -- An enhanced Interactive Python. Type '?' for help.

1]: import torch

2]: torch.cuda.is_available()
2]: True

3]: import tensorflow as tf

4]: tf.config.list_physical_devices('GPU')
4]: []

5]: tf.config.list_physical_devices()
5]: [PhysicalDevice(name='/physical_device:CPU:0', device_type='CPU')]

6]: 
```

# 11 Support

## 11.1 Technical Support

For ADRF technical support, please email support@coleridgeinitiative.org

# References

1. Confidential Information Protection and Statistical Efficiency Act of 2002. (Washington, DC: U.S. GPO, 2002).

2. Federal Committee on Statistical Methodology. "Report on Statistical Disclosure Limitation Methodology," 22 (Second Version, 2005). https://nces.ed.gov/fcsm/pdf/spwp22.pdf.

3. "How to Use Microdata Properly: Self-Study Material for the Users of Eurostat Microdata Sets." (2018). https://ec.europa.eu/eurostat/web/microdata/overview/self-study-material-for-microdata-users.

4. Research Data Centre of the German Federal Employment Agency at the Institute for Employment Research. "Remote Data Access and On-Site Use at the FDZ of the BA at the IAB." (2020, December 8). http://doku.iab.de/fdz/access/Vorgaben_DAFE_EN.PDF

5. Welpton, Richard. Handbook on Statistical Disclosure Control for Outputs. (figshare, 2019). https://doi.org/10.6084/m9.figshare.9958520.v1.

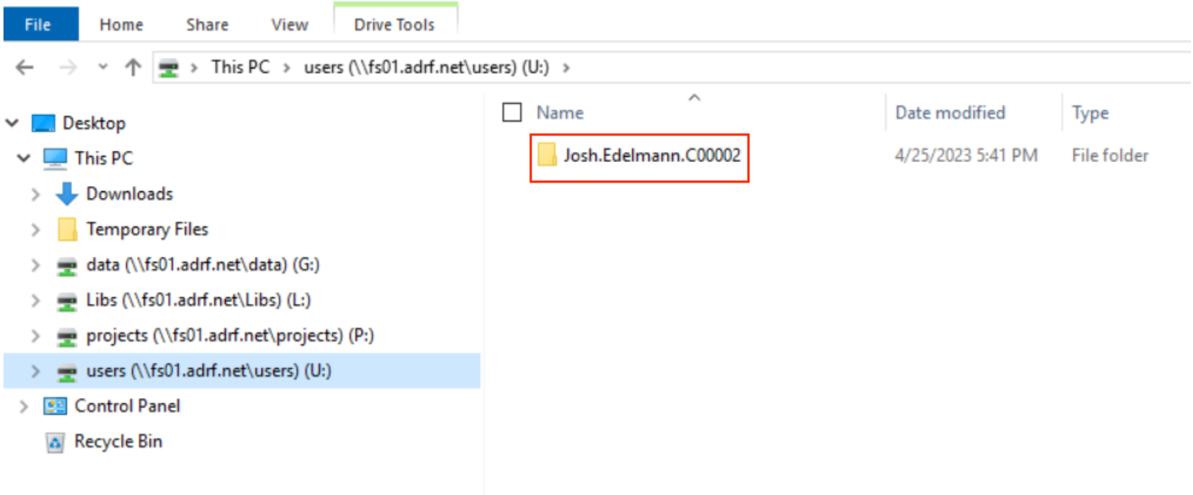
# 12 Redshift querying guide

## 12.1 Introduction

This document serves as an introduction to generating proficient Amazon Redshift queries. This is a generalized document meaning you will need to replace “schema_name” and “table_name” with the appropriate schema and table names used for your project.

## 12.2 Data Access

The data is housed in Redshift. **You need to replace the “user.name.project” with your project based username. The project based username is your user folder name in the `U:/` drive:**



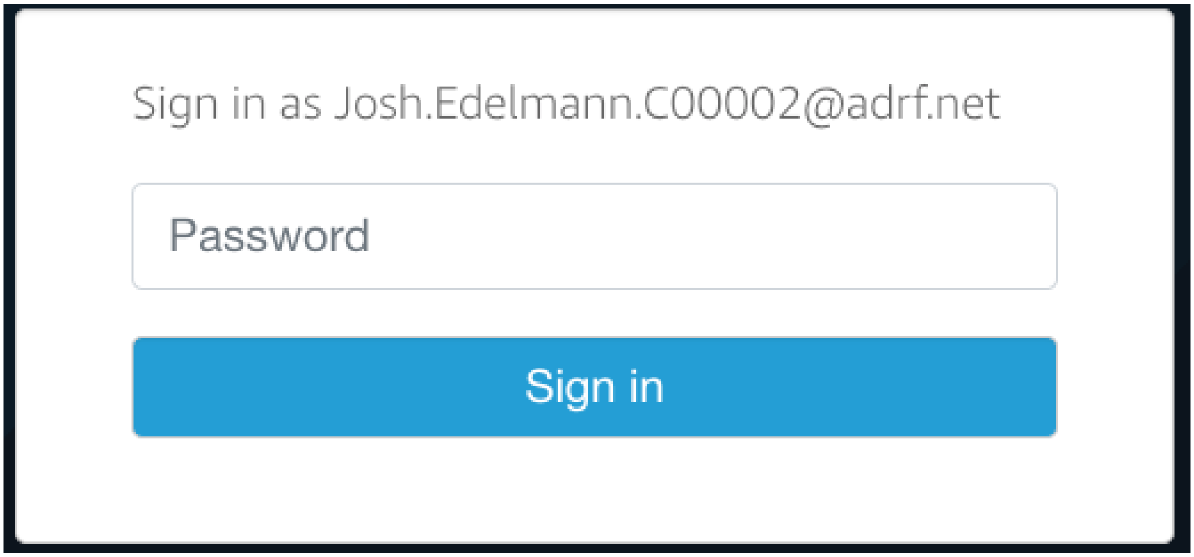
File
Home
Share
View
Drive Tools
This PC › users (\\fs01.adrf.net\users) (U:) ›
Desktop
This PC
Downloads
Temporary Files
data (\\fs01.adrf.net\data) (G:)
Libs (\\fs01.adrf.net\Libs) (L:)
projects (\\fs01.adrf.net\projects) (P:)
users (\\fs01.adrf.net\users) (U:)
Control Panel
Recycle Bin

| □ Name | Date modified | Type |
|---|---|---|
| Josh.Edelmann.C00002 | 4/25/2023 5:41 PM | File folder |

Note: Your username will be different than in these examples.

The password needed to access Redshift is the second password entered when logging into the ADRF as shown in the screen below:

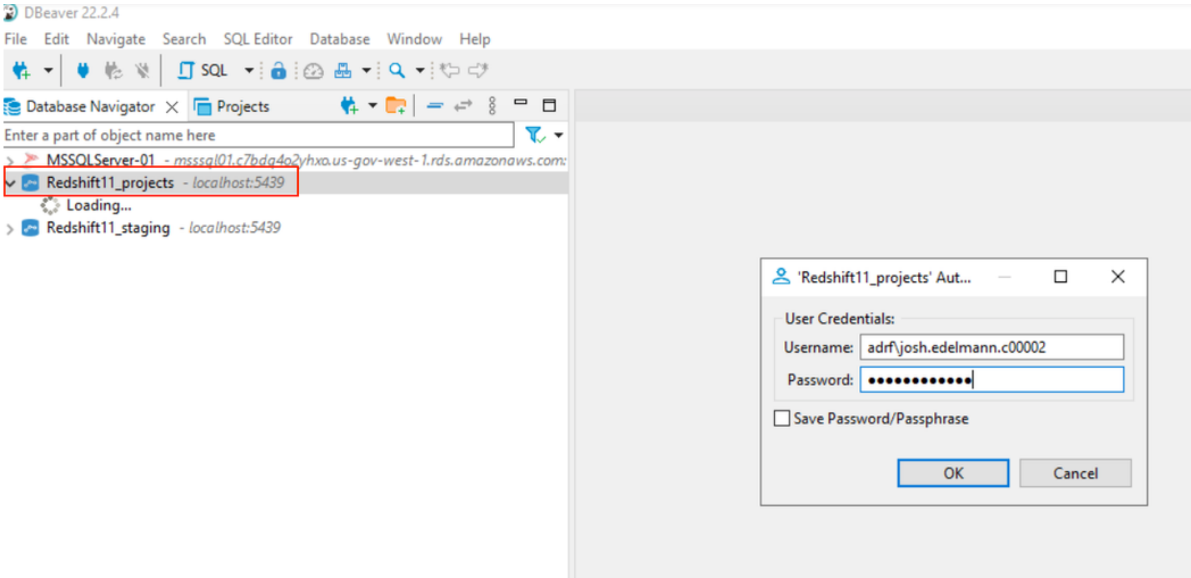
All data is stored under schemas in the ***projects*** database and are accessible by the following programs:



Sign in as Josh.Edelmann.C00002@adrf.net

- **DBeaver**

To establish a connection to Redshift in DBeaver, first double click on the server you wish to connect to. In the example below I'm connecting to `Redshift11_projects`. Then a window will appear asking for your Username and Password. This will be your user folder name and include `adrf\` before the username. Then click OK. You will now have access to your data stored on the `Redshift11_projects` server.



DBeaver 22.2.4
File
Edit
Navigate
Search
SQL Editor
Database
Window
Help
SQL
Database Navigator
Projects
Enter a part of object name here
MSSQLServer-01 - msssql01.c7bdq4o2yhxo.us-gov-west-1.rds.amazonaws.com:
Redshift11_projects - localhost:5439
Loading...
Redshift11_staging - localhost:5439
'Redshift11_projects' Aut...
User Credentials:
□ Save Password/Passphrase
OK
Cancel

Username: adrf\josh.edelmann.c00002

Password: ••••••••••••

### Creating Tables in PR/TR Schema

When users create tables in their PR (Research Project) or TR (Training Project) schema, the table is initially permissioned to the user only. This is analogous to creating a document or file in your U drive: Only you have access to the newly created table.

If you want to allow all individuals in your project workspace to access the table in the PR/TR schema, you will need to grant permission to the table to the rest of the users who have access to the PR or TR schema.

You can do this by running the following code:

```
GRANT SELECT, UPDATE, DELETE, INSERT ON TABLE schema_name.table_name TO group
db_xxxxxx_rw;
```

> Note: In the above code example replace `schma_name` with the `pr_` or `tr_` schema assigned to your workspace and replace `table_name` with the name of the table on which you want to grant access. Also, in the group name `db_xxxxxx_rw`, replace `xxxxxx` with your project code. This is the last 6 characters in your project based user name. This will start with either a `T` or a `P`.

If you want to allow only a single user on your project to access the table, you will need to grant permission to that user. You can do this by running the following code:

```
GRANT SELECT, UPDATE, DELETE, INSERT ON TABLE       schema_name.table_name to
"IAM:first_name.last_name.project_code";
```

> Note: In the above code example replace `schma_name` with the `pr_` or `tr_` schema assigned to your workspace and replace `table_name` with the name of the table on which you want to grant access. Also, in `"IAM:first_name.last_name.project_code"` update `first_name.last_name.project_code` with the user name to whom you want to grant access to.

If you have any questions, please reach out to us at support@coleridgeinitiative.org

When connecting to the database through SAS, R, Stata, or Python you need to use one of the following DSNs:

- **Redshift01_projects_DSN**
- **Redshift11_projects_DSN**

In the code examples below, the default DSN is `Redshift01_projects_DSN`.

- **SAS Connection**

```
proc sql;
connect to odbc as my con
(datasrc=Redshift01_projects_DSN user=adrf\user.name.project password=password);
```

```
select * from connection to mycon
(select * form projects.schema.table);
disconnect from mycon;
quit;
```

- **R Connection**

### Best practices for loading large amounts of data in R

To ensure R can efficiently manage large amounts of data, please add the following lines of code to your R script before any packages are loaded:

`options(java.parameters = c("-XX:+UseConcMarkSweepGC", "-Xmx8192m")) gc()`

### ***Redshift R database connectivity Changes***

*When connecting to Redshift database using R (whether in R Studio or Jupyter Notebook), We strongly recommend that you use a JDBC based connection (versus using an ODBC based connection). Other than how you connect to database, rest of your code should remain the same.*

### Best practices for writing tables to Redshift

When writing tables to Redshift from R in either a SQL query or R data frame, please use the following lines of R code:

Writing a table from R to Redshift using SQL `INTO` statement use the function `dbSendUpdate()`:

`dbSendUpdate(conn, SELECT col_1 INTO schema_name.table_name FROM schema_name.old_table_name`

When writing an R data frame to Redshift use the following code as an example:

```
qry <- "set search_path to schema_name"
dbSendUpdate(conn, qry)

dbWriteTable(
conn = conn, #name of the connection
name = 'table_name', #name of table to save df to
value = df_name, #name of df to write to Redshift
overwrite = TRUE #if you want to overwrite a current table, otherwise FALSE
```

### ***The below table is for connecting to RedShift11 Database***

| | ODBC Based Connection | JDBC Based Connection(Recommended) |
|---|---|---|
| ***Libraries*** | `library(odbc)` | `library(RJDBC)` |

|  |  |  |
|---|---|---|
| ***User ID and Password*** |  | dbusr=Sys.getenv("DBUSER")<br>dbpswd=Sys.getenv("DBPASSWD") |
|  |  | # Database URL<br>url <-<br>"jdbc:redshift:iam://adrf-redshift11.cdy8ch2udkt<br>loginToRp=urn:amazon:webservices:govcloud;<br>ssl=true;<br>AutoCreate=true;<br>idp_host=adfs.adrf.net;<br>idp_port=443;<br>ssl_insecure=true;<br>plugin_name=com.amazon.redshift.plugin.AdfsCrede<br># Redshift JDBC Driver Setting<br>driver <-<br>JDBC("com.amazon.redshift.jdbc42.Driver",<br>classPath =<br>"C:\\drivers\\redshift_withsdk\\redshift-jdbc42-<br>identifier.quote="`") |
| ***Connection*** | con <-<br>dbConnect(odbc(),"Redshift11_projects_DSN", uid<br>= "adrf\\John.doe.p00002", pwd<br>= 'xxxxxxxxxxxxxx') | conn <- dbConnect(driver, url,<br>dbusr, dbpswd) |

*For the above code to work, please create a file name* ***.Renviron*** *in your user folder (user folder is something like i.e. u:\ John.doe.p00002) And* ***.Renviron file*** *should contain the following:*

```
DBUSER='adrf\John.doe.p00002'
DBPASSWD='xxxxxxxxxxxx'
```

*PLEASE replace user id and password with your project workspace specific user is and password.*

*This will ensure you don't have your id and password in R code and then you can easily share your R code with others without sharing your ID and password.*

***The below table is for connecting to RedShift01 Database***

| | *ODBC Based Connection* | *JDBC Based Connection (Recommended)* |
|---|---|---|
| ***Libraries*** | library(odbc) | library(RJDBC) |
| ***User ID and Password*** | | dbusr=Sys.getenv("DBUSER")<br>dbpswd=Sys.getenv("DBPASSWD")<br><br># Database URL<br>url <-<br>"jdbc:redshift:iam://adrf-redshift01.cdy8ch2udkt<br>loginToRp=urn:amazon:webservices:govcloud;<br>ssl=true;AutoCreate=true;<br>idp_host=adfs.adrf.net;<br>idp_port=443;<br>ssl_insecure=true;<br>plugin_name=com.amazon.redshift.plugin.AdfsCrede<br># Redshift JDBC Driver Setting<br>driver <-<br>JDBC("com.amazon.redshift.jdbc42.Driver",<br>classPath =<br>"C:\\drivers\\redshift_withsdk\\redshift-jdbc42-<br>identifier.quote="`") |
| ***Connection*** | con <- dbConnect(odbc(), "Redshift11_projects_DSN", uid = "adrf\\John.doe.p00002", pwd = 'xxxxxxxxxxxxxx') | conn <- dbConnect(driver, url, dbusr, dbpswd) |

*For the above code to work, please create a file name* ***.Renviron*** *in your user folder (user folder is something like  i.e. u:\ John.doe.p00002) And* ***.Renviron file*** *should contain the following:*

```
DBUSER='adrf\John.doe.p00002’
DBPASSWD='xxxxxxxxxxxx'
```

*PLEASE replace user id and password with your project workspace specific user is and password.*

*This will ensure you don’t have your id and password in R code and then you can easily share your R code with others without sharing your ID and password.*

- **Python Connection**

```
import pyodbc
import pandas as pd
cnxn = pyodbc.connect('DSN=Redshift01_projects_DSN;
                       UID = adrf\\user.name.project; PWD = password')
df = pd.read_sql("SELECT * FROM projects.schema_name.table_name", cnxn)
```

- **Stata Connection**

```
odbc load, exec("select * from PATH_TO_TABLE") clear
dsn("Redshift01_projects_DSN") user("adrf\user.name.project) password("password")
```

## 12.3 Redshift Query Guidelines for Researchers

*Developing your query.* Here's an example workflow to follow when developing a query.

1. Study the column and table metadata, which is accessible via the table definition. Each table definition can be displayed by clicking on the [+] next the table name.
2. To get a feel for a table's values, SELECT * from the tables you're working with and LIMIT your results (Keep the LIMIT applied as you refine your columns) or use (e.g., select  * from [table name] LIMIT 1000 )
3. Narrow down the columns to the minimal set required to answer your question.
4. Apply any filters to those columns.
5. If you need to aggregate data, aggregate a small number of rows
6. Once you have a query returning the results you need, look for sections of the query to save as a Common Table Expression (CTE) to encapsulate that logic.

### 12.3.1 *DO and DON'T DO BEST PRACTICES:*

#### 12.3.1.1 Tip 1: Use SELECT <columns> instead of SELECT *

Specify the columns in the SELECT clause instead of using SELECT *. The unnecessary columns place extra load on the database, which slows down not just the single Amazon Redshift, but the whole system.

**Inefficient**

```
SELECT * FROM projects.schema_name.table_name
```

This query fetches all the data stored in the table you choose which might not be required for a particular scenario.

**Efficient**

```
SELECT col_A, col_B, col_C FROM projects.schema_name.table_name
```

#### 12.3.1.2 Tip 2: Always fetch limited data and target accurate results

Lesser the data retrieved, the faster the query will run. Rather than applying too many filters on the client-side, filter the data as much as possible at the server. This limits the data being sent on the wire and you'll be able to see the results much faster. In Amazon Redshift use **LIMIT (###)** qualifier at the end of the query to limit records.

```
SELECT  col_A, col_B, col_C FROM projects.schema_name.table_name  WHERE
[apply some filter] LIMIT 1000
```

#### 12.3.1.3 Tip 3: Use wildcard characters wisely

Wildcard characters can be either used as a prefix or a suffix. Using leading wildcard (%) in combination with an ending wildcard will search all records for a match anywhere within the selected field.

**Inefficient**

```
Select col_A, col_B, col_C from projects.schema_name.table_name where col_A
like '%BRO%'
```

This query will pull the expected results of **Brown Sugar, Brownie, Brown Rice** and so on. However, it will also pull unexpected results, such as **Country Brown, Lamb with Broth, Cream of Broccoli**.

**Efficient**

```
Select col_A, col_B, col_C from projects.schema_name.table_name where col_B
like 'BRO%'.
```

This query will pull only the expected results of **Brownie, Brown Rice, Brown Sugar** and so on.

#### 12.3.1.4 Tip 4: Does My record exist?

Normally, developers use EXISTS() or COUNT() queries for matching a record entry. However, EXISTS() is more efficient as it will exit as soon as finding a matching record; whereas, COUNT() will scan the entire table even if the record is found in the first row.

**Efficient**

```
select col_A from projects.schema_name.table_name A where exists (select 1
from projects.schema_name.table_name B where A.col_A = B.col_A ) order by
col_A;
```

#### 12.3.1.5 Tip 5: Avoid correlated subqueries

A correlated subquery depends on the parent or outer query. Since it executes row by row, it decreases the overall speed of the process.

**Inefficient**

```
SELECT col_A, col_B, (SELECT col_C FROM projects.schema_name.table_name_a
WHERE col_C = c.rma LIMIT 1) AS new_name FROM projects.schema_name.table_name_b
```

Here, the problem is — the inner query is run for each row returned by the outer query. Going over the **"table_name_b"** table again and again for every row processed by the outer query creates process overhead. Instead, for Amazon Redshift query optimization, use JOIN to solve such problems.

**Efficient**

```
SELECT col_A, col_B, col_C FROM projects.schema_name.table_name c LEFT JOIN
projects.schema_name.table_name co ON c.col_A = co.col_B
```

#### 12.3.1.6 Tip 6: Avoid using Amazon Redshift function in the where condition

Often developers use functions or methods with their Amazon Redshift queries.

**Inefficient**

```
SELECT col_A, col_B, col_C FROM projects.schema_name.table_name WHERE RIGHT(birth_date,4)
= '1965' and  LEFT(birth_date,2) = '07'
```

Note that even if **birth_date** has an index, the above query changes the WHERE clause in such a way that this index cannot be used anymore.

**Efficient**

```
SELECT col_A, col_B, col_C FROM projects.schema_name.table_name WHERE birth_date
between '711965' and '7311965'
```

#### 12.3.1.7 Tip 7: Use WHERE instead of HAVING

HAVING clause filters the rows after all the rows are selected. It is just like a filter. Do not use the HAVING clause for any other purposes. It is useful when performing group bys and aggregations.

#### 12.3.1.8 Tip 8: Use temp tables when merging large data sets

Creating local temp tables will limit the number of records in large table joins and merges. Instead of performing large table joins, one can break out the analysis by performing the analysis in two steps: 1) create a temp table with limiting criteria to create a smaller / filtered result set. 2) join the temp table to the second large table to limit the number of records being fetched and to speed up the query. This is especially useful when there are no indexes on the join columns.

**Inefficient**

```
SELECT col_A, col_B, sum(col_C) total FROM projects.schema_name.table_name  pd  INNER
JOIN  projects.schema_name.table_name st  ON  pd.col_A=st.col_B WHERE pd.col_C
like 'DOG%' GROUP BY pd.col_A, pd.col_B, pd.col_C
```

Note that even if joining column **col_A** has an index, the col_B column does not. In addition, because the size of some tables can be large, one should limit the size of the join table by first building a smaller filtered #temp table then performing the table joins.

**Efficient**

```
SET search_path = schema_name;  -- this statement sets the default schema/database
to projects.schema_name
```

**Step 1:**

```
CREATE TEMP TABLE temp_table (

col_A varchar(14),

col_B varchar(178),

col_C varchar(4) );
```

**Step 2:**

```
INSERT INTO temp_table SELECT col_A, col_B, col_C

FROM projects.schema_name.table_name WHERE col_B like 'CAT%';
```

**Step 3:**

```
SELECT pd.col_A, pd.col_B, pd.col_C, sum(col_C) as total FROM  temp_table
pd  INNER JOIN  projects.schema_name.table_name st  ON pd.col_A=st.col_B  GROUP
BY pd.col_A, pd.col_B, pd.col_C;
```

```
DROP TABLE temp_table;
```

Note always drop the temp table after the analysis is complete to release data from physical memory.

### 12.3.2 *Other Pointers for best database performance*

***SELECT columns, not stars.*** Specify the columns you'd like to include in the results (though it's fine to use * when first exploring tables — just remember to LIMIT your results).

***Avoid using SELECT DISTINCT***. SELECT DISTINCT command in Amazon Redshift used for fetching unique results and remove duplicate rows in the relation. To achieve this task, it basically groups together related rows and then removes them. GROUP BY operation is a costly operation. To fetch distinct rows and remove duplicate rows, use more attributes in the SELECT operation.

***Inner joins vs WHERE clause.*** Use inner join for merging two or more tables rather than using the WHERE clause. WHERE clause creates the CROSS join/ CARTESIAN product for merging tables. CARTESIAN product of two tables takes a lot of time.

***IN versus EXISTS.*** IN operator is costlier than EXISTS in terms of scans especially when the result of the subquery is a large dataset. We should try to use EXISTS rather than using IN for fetching results with a subquery.

**Avoid**

```
SELECT col_A , col_B, col_C

FROM projects.schema_name.table_name

WHERE col_A IN

(SELECT col_B FROM projects.schema_name.table_name WHERE col_B = 'DOG')
```

**Prefer**

```
SELECT col_A , col_B, col_C

FROM projects.schema_name.table_name

WHERE EXISTS

(SELECT col_A FROM projects.schema_name.table_name b WHERE
a.col_A = b.col_B and b.col_B = 'DOG')
```

Query optimizers can change the order of the following list, but this general lifecycle of a Amazon Redshift query is good to keep in mind when writing Amazon Redshift.

1. **FROM** (and JOIN) get(s) the tables referenced in the query.
2. **WHERE** filters data.
3. **GROUP BY** aggregates data.
4. **HAVING** filters out aggregated data that doesn't meet the criteria.
5. **SELECT** grabs the columns (then deduplicates rows if DISTINCT is invoked).
6. **UNION** merges the selected data into a result set.
7. **ORDER BY** sorts the results.

### 12.3.3 Amazon Redshift best practices for FROM

*Join tables using the ON keyword.* Although it's possible to "join" two tables using a WHERE clause, use an explicit JOIN. The JOIN + ON syntax distinguishes joins from WHERE clauses intended to filter the results.

```
SET search_path = schema_name;-- this statement sets the default schema/database to projects.schema_name

SELECT A.col_A , B.col_B, B.col_C

FROM projects.schema_name.table_name as A

JOIN projects.schema_name.table_name B ON A.col_A = B.col_B
```

*Alias multiple tables.* When querying multiple tables, use aliases, and employ those aliases in your select statement, so the database (and your reader) doesn't need to parse which column belongs to which table.

**Avoid**

```
SET search_path = schema_name;-- this statement sets the default schema/database to projects.schema_name

SELECT col_A , col_B, col_C

FROM dbo.table_name as A

LEFT JOIN dbo.table_name as B ON A.col_A = B.col_B
```

**Prefer**

```
SET search_path = schema_name;-- this statement sets the default schema/database to projects.schema_name

SELECT A.col_A , B.col_B, B.col_C

FROM dbo.table_name as A

LEFT JOIN dbo.table_name as B

A.col_A = B.col_B
```

### 12.3.4 Amazon Redshift best practices for WHERE

*Filter with WHERE before HAVING.* Use a WHERE clause to filter superfluous rows, so you don't have to compute those values in the first place. Only after removing irrelevant rows, and after aggregating those rows and grouping them, include a HAVING clause to filter out aggregates.

*Avoid functions on columns in WHERE clauses.* Using a function on a column in a WHERE clause can really slow down your query, as the function prevents the database from using an index to speed up the query. Instead of using the index to skip to the relevant rows, the function on the column forces the database to run the function on each row of the table. The concatenation operator || is also a function, so don't try to concat strings to filter multiple columns. Prefer multiple conditions instead:

**Avoid**

```
SELECT col_A, col_B, col_C FROM projects.schema_name.table_name

WHERE concat(col_A, col_B) = 'REGULARCOFFEE'
```

**Prefer**

```
SELECT col_A, col_B, col_C FROM projects.schema_name.table_name

WHERE col_A ='REGULAR' and col_B = 'COFFEE'
```

### 12.3.5 Amazon Redshift best practices for GROUP BY

*Order multiple groupings by descending cardinality.* Where possible, GROUP BY columns in order of descending cardinality. That is, group by columns with more unique values first (like IDs or phone numbers) before grouping by columns with fewer distinct values (like state or gender).

### 12.3.6 Amazon Redshift best practices for HAVING

*Only use HAVING for filtering aggregates.* Before HAVING, filter out values using a WHERE clause before aggregating and grouping those values.

```
SELECT  col_A, sum(col_B) as total_amt

FROM  projects.schema_name.table_name

WHERE  col_C = 1617 and col_A='key'

GROUP BY col_A

HAVING  sum(col_D)> 0
```

### 12.3.7 Amazon Redshift best practices for UNION

*Prefer UNION All to UNION.* If duplicates are not an issue, UNION ALL won't discard them, and since UNION ALL isn't tasked with removing duplicates, the query will be more efficient

### 12.3.8 Amazon Redshift best practices for ORDER BY

*Avoid sorting where possible, especially in subqueries.* If you must sort, make sure your subqueries are not needlessly sorting data.

**Avoid**

```
SELECT col_A, col_B, col_C

FROM projects.schema_name.table_name

WHERE col_B IN

(SELECT col_A  FROM projects.schema_name.table_name

WHERE col_C = 534905 ORDER BY col_B);
```

**Prefer**

```
SELECT col_A, col_B, col_C

FROM projects.schema_name.table_name

WHERE col_A IN

(SELECT col_B  FROM projects.schema_name.table_name

WHERE col_C = 534905);
```

### 12.3.9 Troubleshooting Queries

There are several metrics for calculating the cost of the query in terms of storage, time, CPU utilization. *However, these metrics require DBA permissions to execute.* Follow up with ADRF support to get additional assistance.

**Using the SVL_QUERY_SUMMARY view:** To analyze query summary information by stream, do the following:

**Step 1:** `select query, elapsed, substring from svl_qlog order by query desc limit 5;`

**Step 2:** `select * from svl_query_summary where query = MyQueryID order by stm, seg, step;`

***Execution Plan:*** Lastly, an execution plan is a detailed step-by-step processing plan used by the optimizer to fetch the rows. It can be enabled in the database using the following procedure:

1. Click on SQL Editor in the menu bar.
2. Click on Explain Execution Plan.

It helps to analyze the major phases in the execution of a query. We can also find out which part of the execution is taking more time and optimize that sub-part. The execution plan shows which tables were accessed, what index scans were performed for fetching the data. If joins are present it shows how these tables were merged. Further, we can see a more detailed analysis view of each sub-operation performed during query execution.

## 12.4 AWS Sources

Amazon Redshift best practices for designing queries - Amazon Redshift

Troubleshooting queries - Amazon Redshift

Amazon Redshift and PostgreSQL - Amazon Redshift

Using the SVL_QUERY_SUMMARY view - Amazon Redshift

**Python:**

Examples of using the Amazon Redshift Python connector - Amazon Redshift

**R:**

[Examples of using R with Amazon Redshift] (https://aws.amazon.com/blogs/big-data/connecting-r-with-amazon-redshift/)

**ODBC configuration (for SAS and Stata):**

Configuring a connection for ODBC driver version 2.x for Amazon Redshift - Amazon Redshift