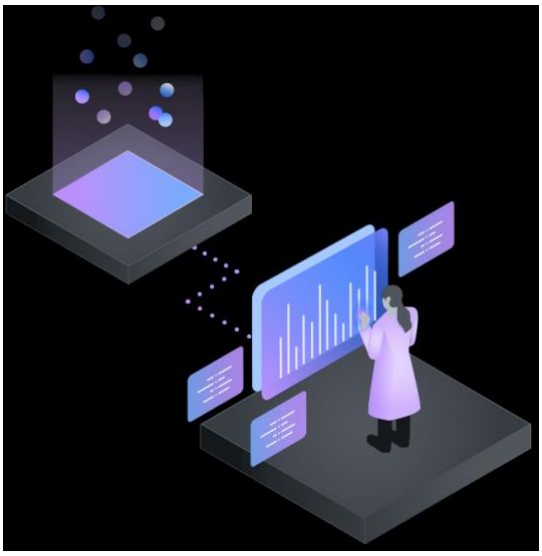
IBM Journey to Cloud and AI

Analytics Modernization Workshop

Featuring: IBM Cloud Pak for Data

Deeper Dive Lab Workbook



Lab workbook and IBM Cloud Pak for Data workshop design and environment by:
Burt Vialpando, IBM Executive Analytics Architect
Kent Rubin, IBM Solution Architect

September 14, 2020

Acknowledgements

- **Duane Almeter** and **Eric Watson** for their leadership on the project
- **Daniel Kikuchi** for the CPD 3.0 cluster install environments, and other technical input and support
- **Ed Duhe, Rich Russo** and **Mitchell Odum** for providing the ESX server development platform
- **John Lucas** for product testing, workbook publishing, and project management support
- **John Van Buren** for the Organize lab designs
- **Rajesh Kartha** for Data Virtualization caching lab and DV z/OS lab
- **Benjamin Herta** for the AutoAI non-AVX work around
- **Sidney Phoon** for the last Notebook updates
- **Eric Martens** for the OpenScale lab design
- **Owais Hashmi** for OpenScale storyboarding and lab assistance
- **Rohit Gargate** for OpenScale Auto setup assistance
- **Ben Chard** for providing Analyze lab resources and assistance
- **Tom Konchan** for SPSS/IPS and Decision Optimization lab
- **Daniel Hancock** for the NPS Getting Started lab
- **David Trotter** for the z/OS work on the DV z/OS lab
- **Linda Snow** for providing assistance with complete workshop review

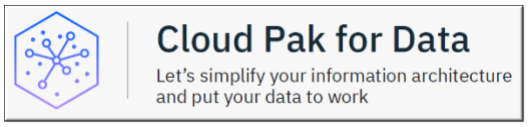


Table of Contents

[This page left intentionally blank]

Lab 11 COLLECT: DATA VIRTUALIZATION CACHING - DEEPER DIVE

11.1 Lab overview

IBM Cloud Pak for Data (CPD) is a robust, fully integrated platform that will improve business agility and decrease the cost of creating high quality, impactful Analytics.

Data Virtualization (DV), an important component of IBM Cloud Pak for Data, helps integrate data sources across multiple types and locations and turn them into one logical data view. This virtual data lake decreases the amount of time needed to get value out of your data. With data virtualization, you can query data across many source systems without having to copy and replicate the sources into a centralized repository, which saves time and reduces cost. It simplifies your analytics and provides “current state of the business” answers because you’re querying the data at its source with all its respective policies enforced. One can create virtual objects in DV, pointing to the original data sources to be used in queries as if they were local objects. Each query executed against these virtual objects gets efficiently compiled and optimized to work directly with the source data. The query times and performance will be dictated by latency and the workload on the underlying data sources, especially while performing aggregated queries across different data sources.

With Data Virtualization Caching, it is possible to avoid trips to the original source(s) and create a temporary cache of the data. This helps immensely with query performance as the DV optimizer can choose to use the local cached copy instead of querying the remote sources repeatedly.

In this lab, you will create Data Virtualization objects that will be used within a dashboard to perform analysis followed by using the caching capability of DV to experience how query times can be drastically reduced. You will also learn how caches can be refreshed, disabled, enabled and deleted, and the best practices and guidelines to create DV Caches.

In our scenario, the Trade Co. Business Analyst (BA) initially creates a dashboard using data from a DV view, which in turn points to multiple remote data sources underneath. But rendering these dashboards and using them for analysis takes a very long time since the data has to be retrieved every time from those remote data sources. The BA works with the DV Admin, who helps create a cache for the virtualized view to help address the data retrieval performance issues observed while working with the dashboard.

11.2 Personas represented in this lab

The Business Analyst persona along with the Data Engineer and Data Virtualization Administrator personas will perform the exercises in this lab.

Persona (Role)	Capabilities
 Business Analyst	Business Analysts deliver value by taking data, using it to answer questions, and communicating the results to help make better business decisions.
 Data Engineer	Data Engineers build and optimize the systems to allow data scientists and business analysts to perform their work. The Data Engineer ensures that any data is properly received, transformed, stored, and made accessible to other users.
 Administrator (DV/CPD)	Administrators set up and maintain the DV module within the CPD environment itself. They are responsible for granting DV access to users and administration tasks such as creating a data cache.

For convenience in doing this lab, instead of switching between personas, all the required privileges have been provided to the same user. The workbook will refer to the respective personas at different stages to help understand the flow of this task.

11.3 Logging into the CPD web client (if you have not already done so)

__1. If you are starting this lab stand-alone (without going through previous labs) do the following:

__2. Double-click the desktop icon: Cloud Pak for Data Web Client.

__3. The CPD web client GUI displays as shown. Use cpduser and cpdaccess for the *Username* and *Password* and click Sign in.

11.4 Reviewing the dashboard: Stock Trading Analysis - Trade Co.

In an attempt to understand stock trading patterns, the business analysis for Trade Co. starts by creating a simple dashboard to find the most popular historically traded stocks. The dashboard shows the number of Shares Sold vs. the number of Daily Trades.

The Business Analyst (BA) works with the Data Engineer (DE) to get virtualized access to the different data sources required for creating the dashboard(s). The DE creates the Data Sources, Virtual Tables and finally a Virtualized View joining all the Virtual Tables and computing the basic aggregations required. The View is then shared with the BA, who can then proceed with creating the dashboard.

However, once the dashboard is initially created, the BA notices delays in rendering it. Since every request from the dashboard has to fetch the data from its original source(s), latency starts to play an important role, slowing down response. The BA works with the Data Virtualization Admin to create a cache for the View, which helps speed up query times significantly and hence rendering of the dashboard(s).

11.5 Virtualizing the remote tables and creating a view

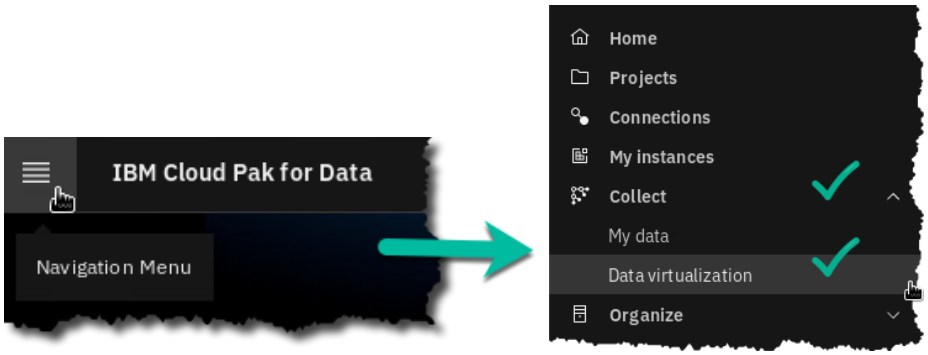
The Data Virtualization process begins by adding data sources to virtualize and is typically done by the Data Engineer.

Persona (Role)
 Data Engineer (DV/CPD)

11.5.1 Navigate to Data virtualization

__4. Start at the Navigation Menu.

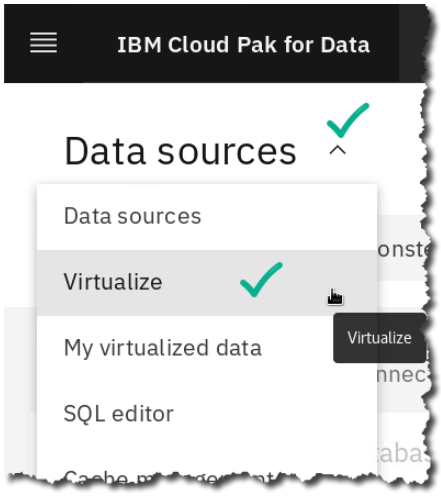
Click Collect ⇨ Data virtualization.



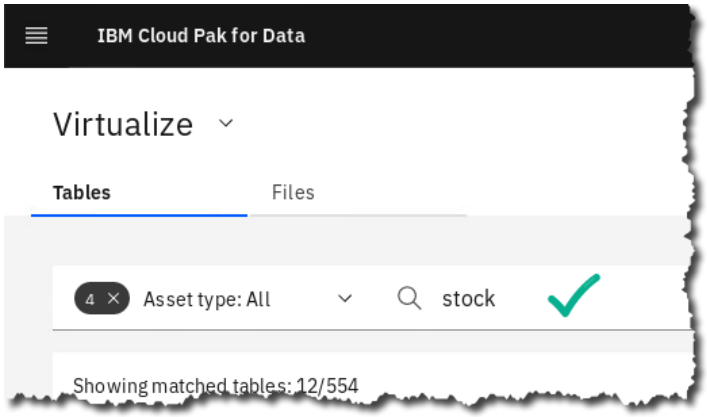
11.5.2 Creating virtualized tables

__5. With the data sources successfully created, the next step is to virtualize the tables needed for this exercise.

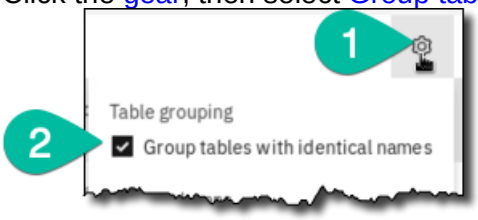
From the top left go to Data Sources ⇨ Virtualize.



__6. In the search bar, enter the string STOCK and click the search icon (magnifying glass).

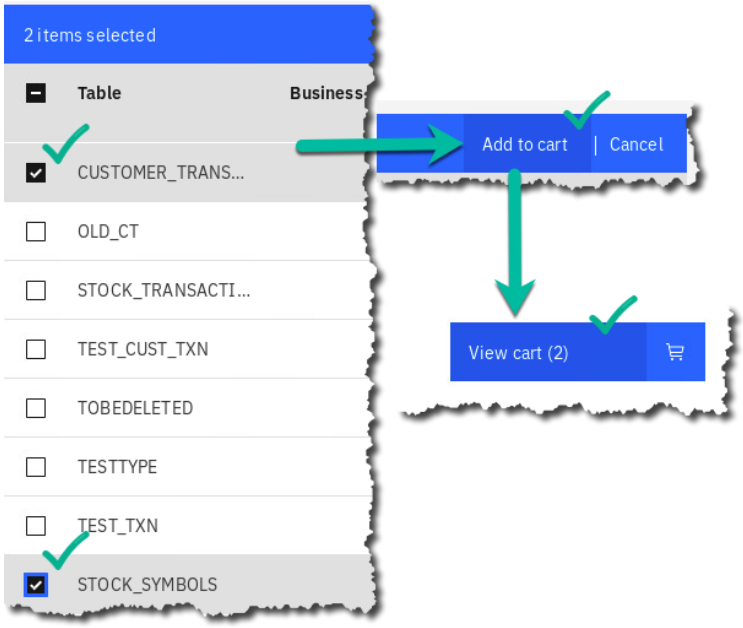


__7. Click the gear, then select Group tables with identical names.

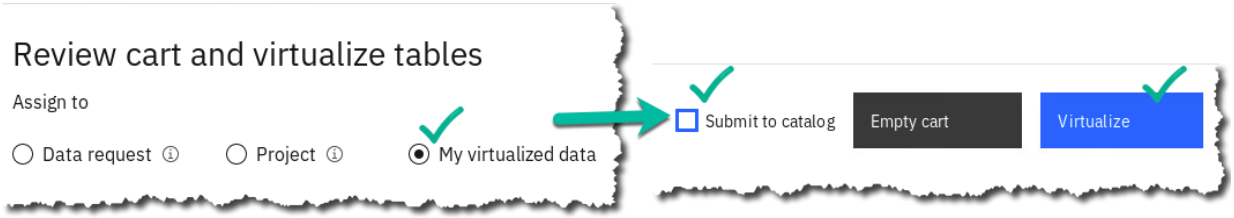


This uses the DV feature of “Schema Folding” which groups tables with the same names across data sources and shows them as a single “table” selection. (Notice how two tables called CUSTOMER_TRANSACTIONS from to different databases are treated as one.)

__8. Select tables: CUSTOMER_TRANSACTIONS and STOCK_SYMBOLS ⇨Click Add to cart ⇨ View cart.

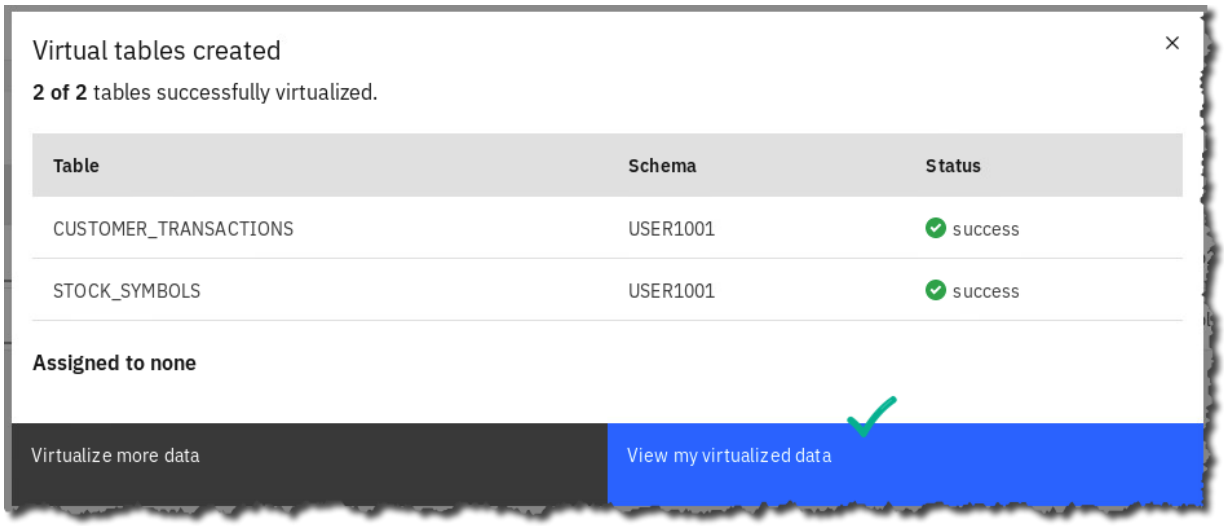


__9. In the Review cart and Virtualize tables section, select button My virtualized data ⇨Uncheck box Submit to catalog ⇨ Click Virtualize.



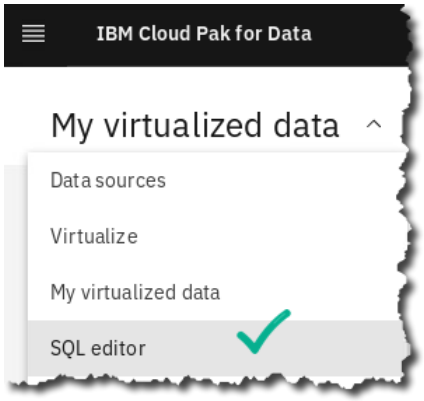
__10. Notice the **two** virtual tables you just created, which are actually comprised of three different tables in three databases in the IBM Cloud. The CUSTOMER_TRANSACTIONS virtual table is a schema folded table pointing to **two** remote Db2 Warehouse tables and the STOCK_SYMBOLS virtual table points to **single** table in the MySQL database.

Click View my virtualized data.



__11. You will now be creating a virtualized view of these virtualized tables using the SQL editor.

Select Menu (My virtualized data) ⇨ SQL editor.

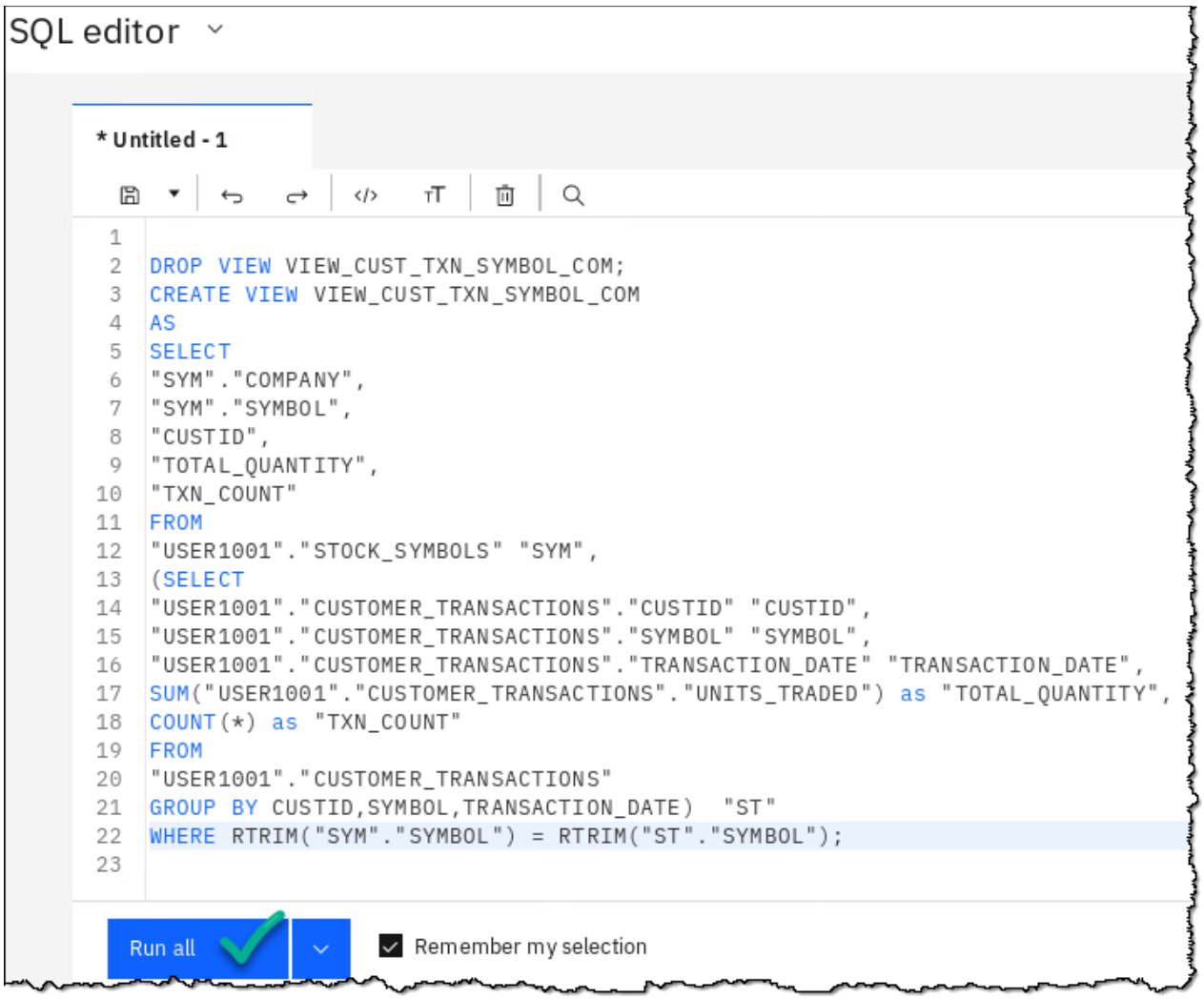


__12. Copy and paste the SQL below into the SQL editor.

 Data Engineer	Note: You can copy and paste this SQL directly from the Unified Desktop by using the File Browser to open this file:  ⇨ Home ⇨ Downloads ⇨ DVCaching_SQL.txt. Alternately, you can download this file by opening a browser tab and using this link: https://ibm.box.com/v/Workshop-V9-DVCaching-SQL

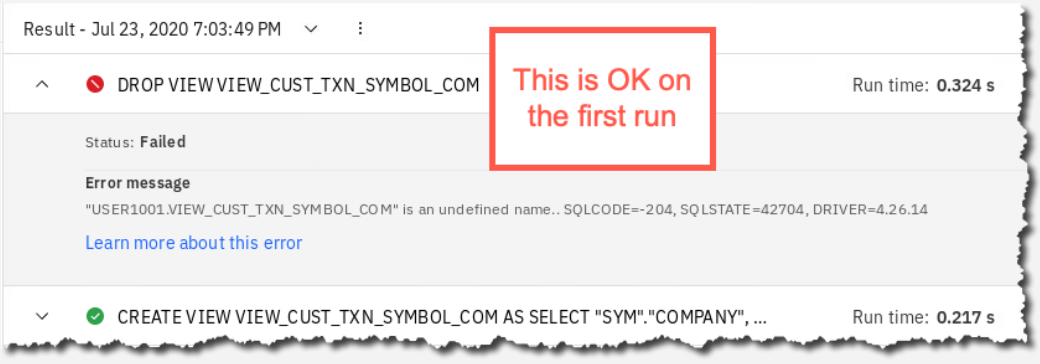
```
DROP VIEW VIEW_CUST_TXN_SYMBOL_COM;
CREATE VIEW VIEW_CUST_TXN_SYMBOL_COM
AS
SELECT
"SYM"."COMPANY",
"SYM"."SYMBOL",
"CUSTID",
"TOTAL_QUANTITY",
"TXN_COUNT"
FROM
"USER1001"."STOCK_SYMBOLS" "SYM",
(SELECT
"USER1001"."CUSTOMER_TRANSACTIONS"."CUSTID" "CUSTID",
"USER1001"."CUSTOMER_TRANSACTIONS"."SYMBOL" "SYMBOL",
"USER1001"."CUSTOMER_TRANSACTIONS"."TRANSACTION_DATE"
"TRANSACTION_DATE",
SUM("USER1001"."CUSTOMER_TRANSACTIONS"."UNITS_TRADED") as
"TOTAL_QUANTITY",
COUNT(*) as "TXN_COUNT"
FROM
"USER1001"."CUSTOMER_TRANSACTIONS"
GROUP BY CUSTID,SYMBOL,TRANSACTION_DATE) "ST"
WHERE RTRIM("SYM"."SYMBOL")= RTRIM("ST"."SYMBOL");
```

__13. Click Run all which should create the virtualized view successfully.



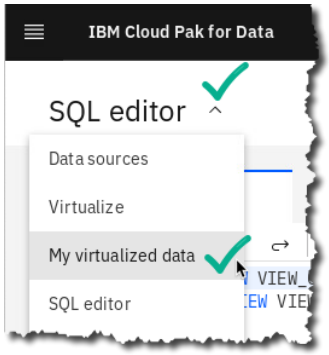
Data Engineer

Note: The first time this script is run, the DROP VIEW statement will fail since the view does not exist. It will execute cleanly on subsequent runs.

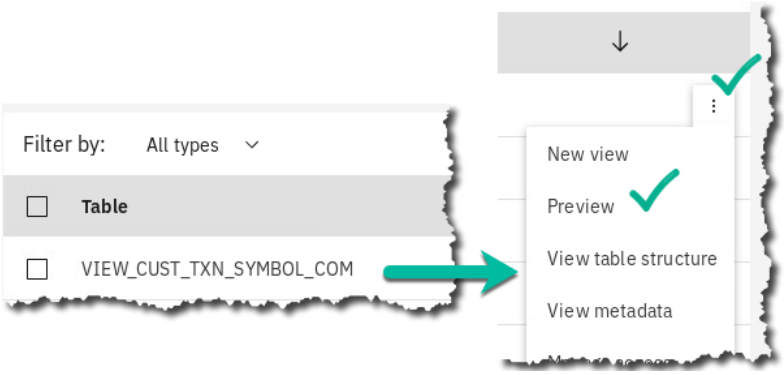


__14. Preview the contents of the virtualized view to confirm the data is being correctly retrieved.

Go to Menu (SQL editor) ⇨ My virtualized data.

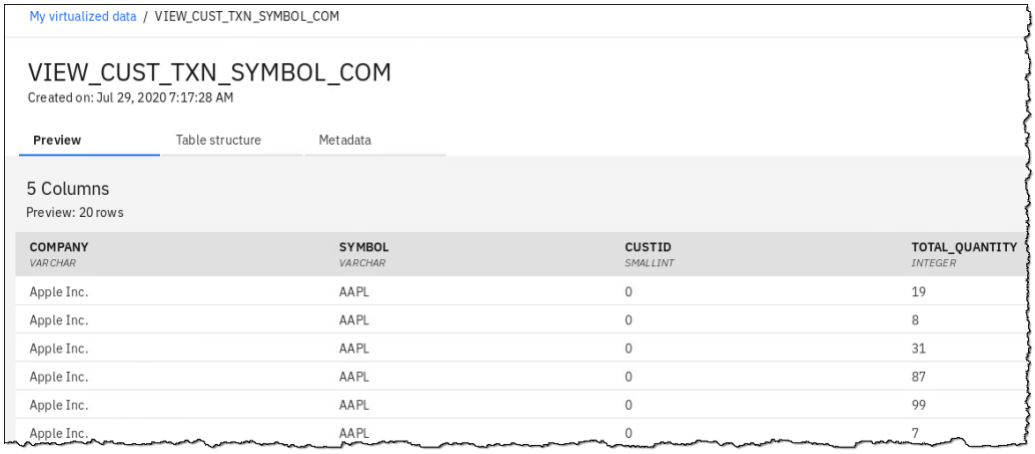


__15. On the view you just created, VIEW_CUST_TXN_SYMBOL_COM, click the ellipses at the end of that line … and then select Preview.



Data Engineer	It may take a minute or two for the view to render. If this step fails, check to make sure that you have copied the SQL code properly into the SQL editor and run it again. Keep in mind that a successful execution of the SQL may not necessarily mean the view was actually created properly.

__16. The Virtualized View preview should render as below.



11.6 Creating a project to work in

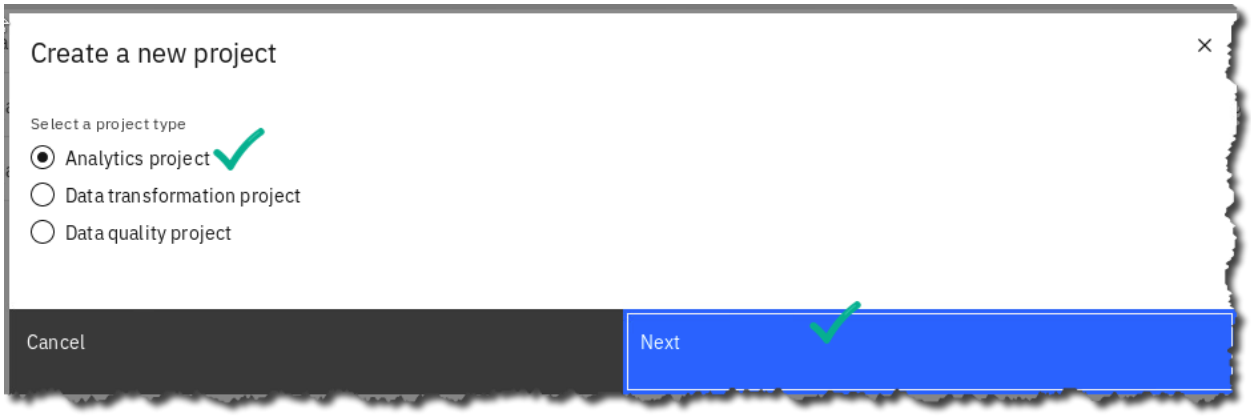
The Virtualized View referencing different data sources is now created. The Data Engineer will share the Virtualized View with the Business Analyst (BA) who can then work on creating the dashboard.

Persona (Role)
 Business Analyst

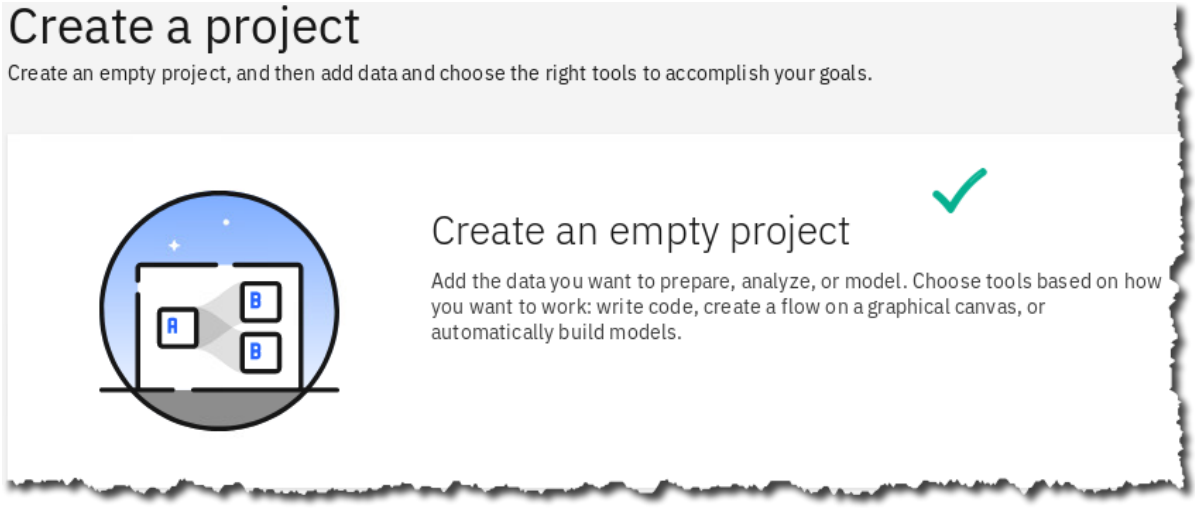
With the access to the Virtualized View in place, the BA begins to work on the dashboard by creating an Analytics Project.

__17. Click Navigation Menu ⇨ Projects ⇨ New project.

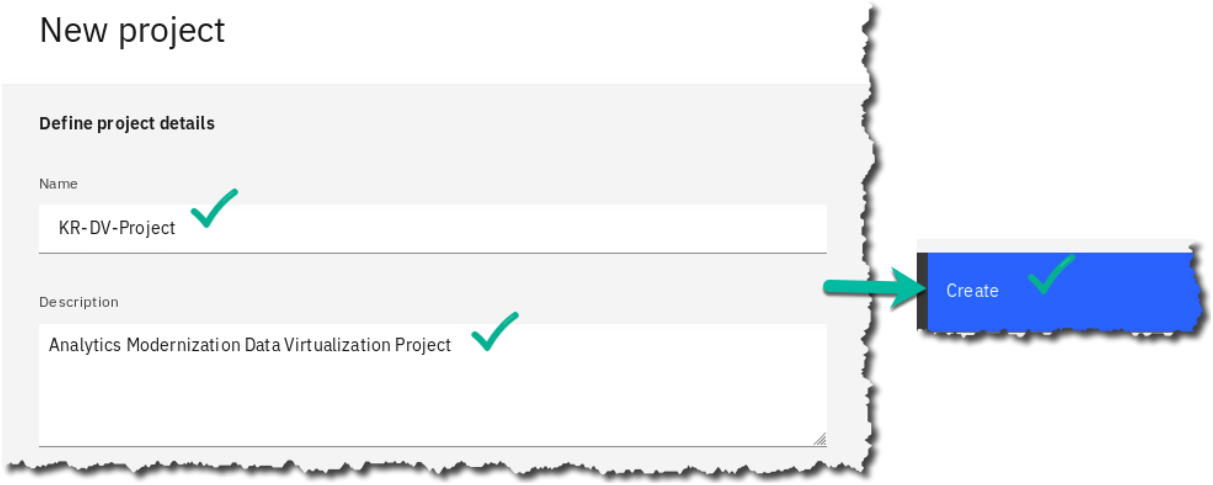
__18. Choose Analytics project ⇨ Next.



__19. Click on tile Create an empty project.

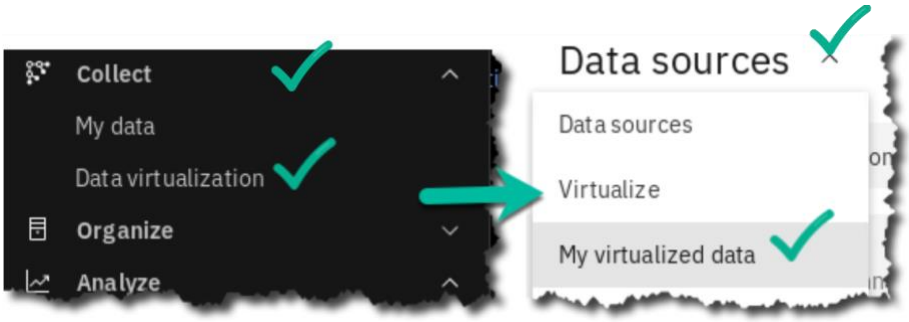


__20. Provide a name of 'your initials-DV-Project' and description for the project and click Create.



11.6.1 Adding the previously created DV view to the new project

__21. Start at the Navigation Menu ⇨ Click Collect ⇨ Data virtualization ⇨ Menu (Data Sources) ⇨ My virtualized data.

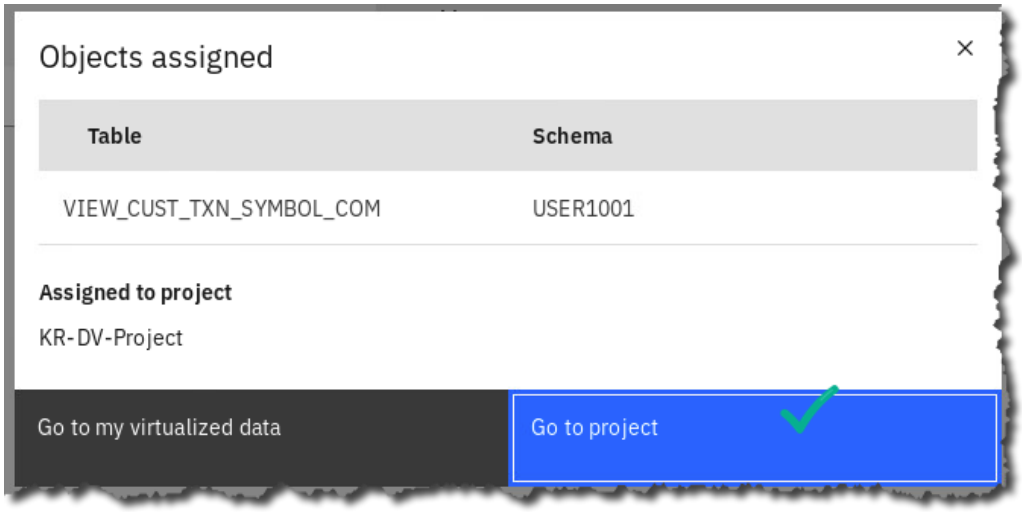


__22. Check VIEW_CUST_TXN_SYMBOL_COM ⇨ click Assign.

__23. Choose your_initials-DV-Project (whatever you have named it) ⇨ Assign.

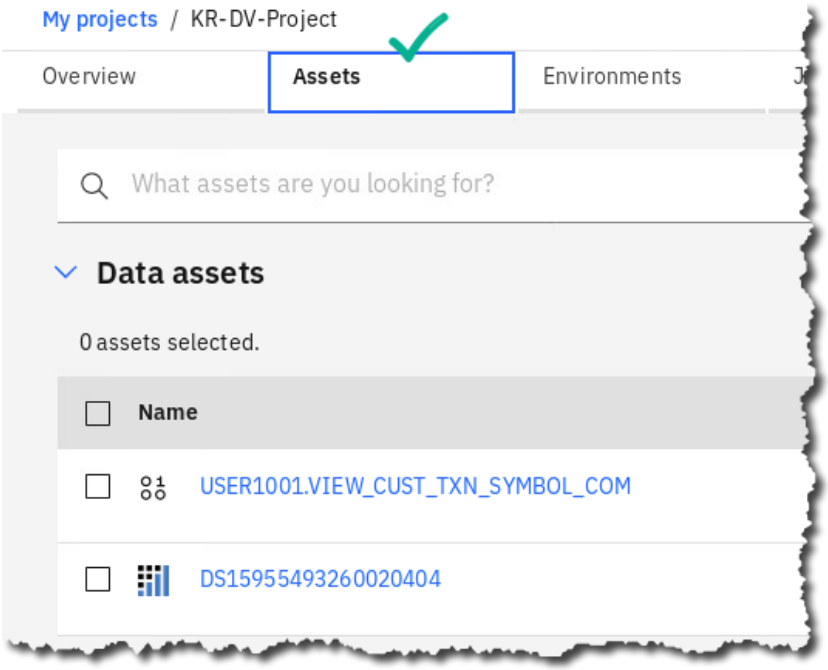
__24. The DV View is now successfully assigned to the project.

Click Go to project.



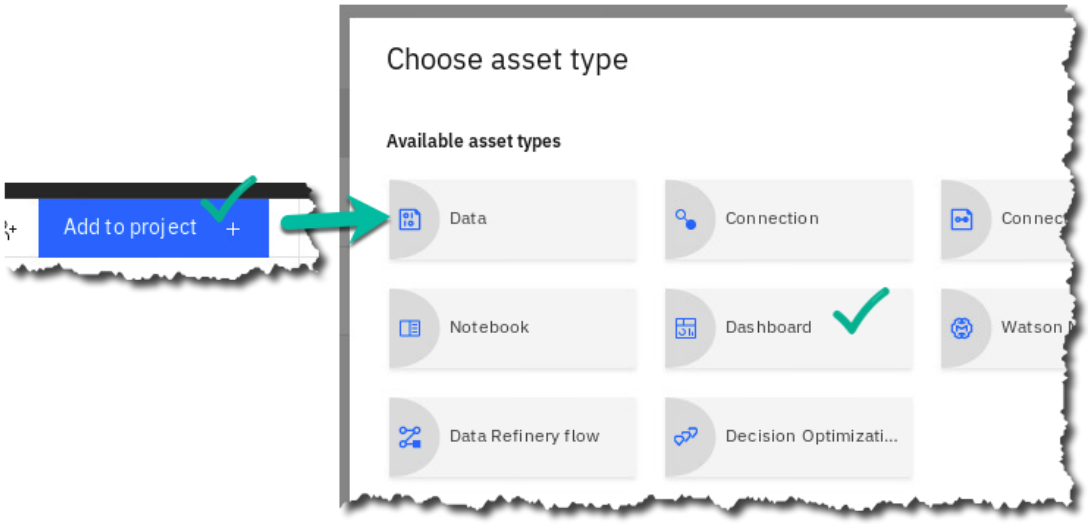
__25. On the Project page, click the Assets tab.

Review the Data assets section where the newly assigned virtualized view should now show up, along with the Connection it needs in order to render it from the data sources. (Your connection Name will vary from what is shown below.)



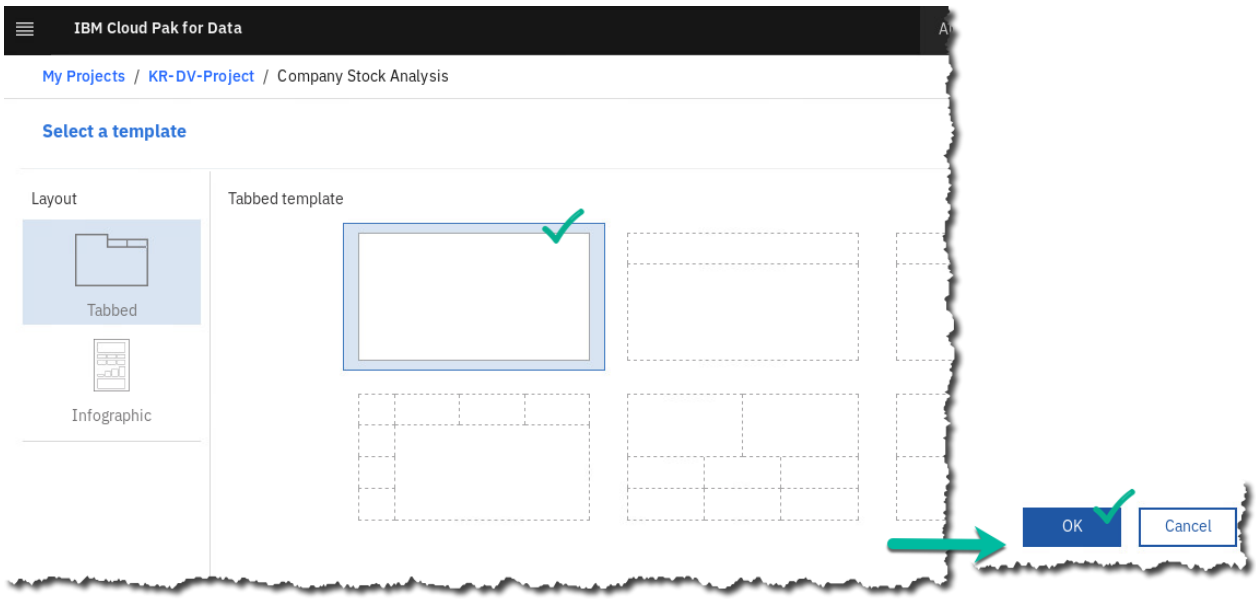
11.6.2 Creating a dashboard within the project to use the data asset:

__26. Click + Add to project and choose Dashboard as the asset type.

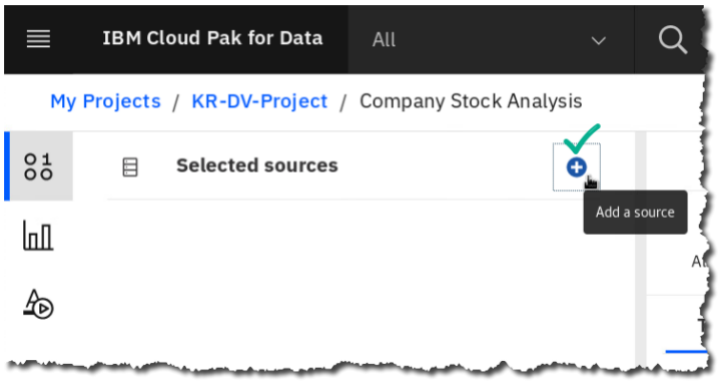


__27. Under *Name* type: Company Stock Analysis ⇨ Create.

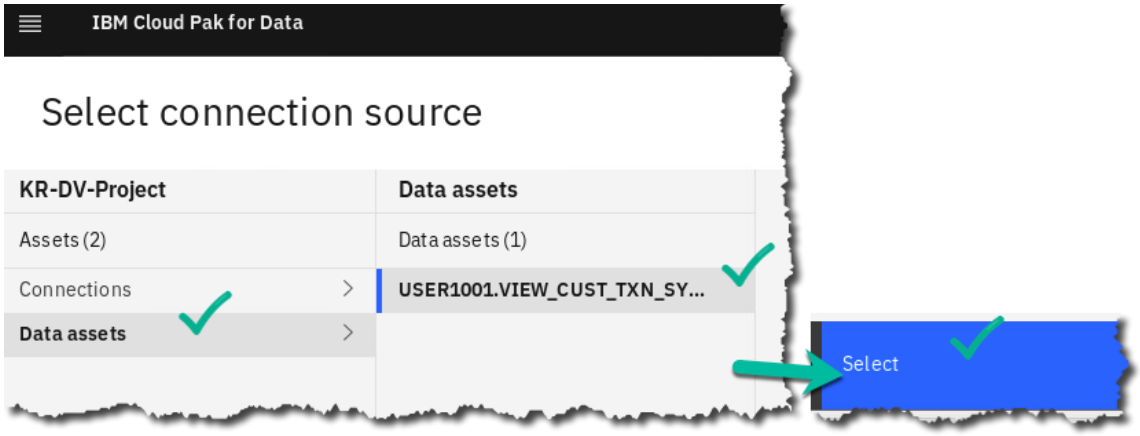
__28. Choose the Freeform (default) layout ⇨ OK.



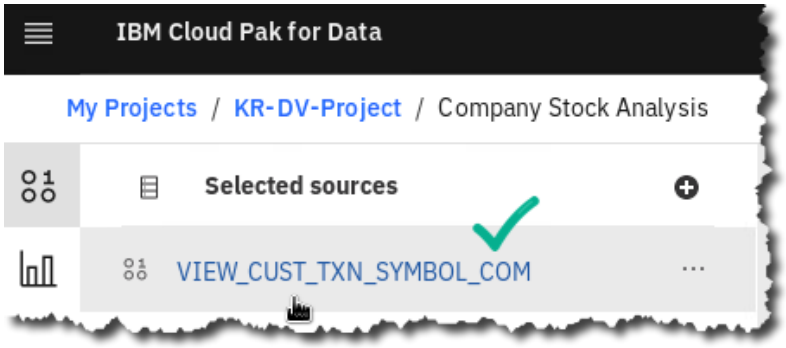
__29. On top left of the dashboard canvas in Selected sources and click + (Add a source).



__30. Click Data Assets ⇨ USER1001.VIEW_CUST_TXN_SYMBOL_COM ⇨ Select.

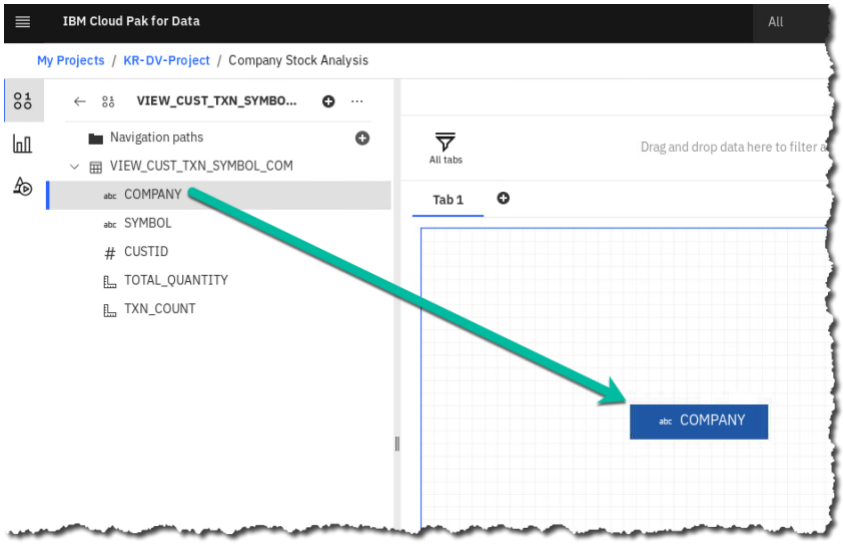


__31. Click the newly added data source to expand the view to see the columns available to be used in the dashboard.



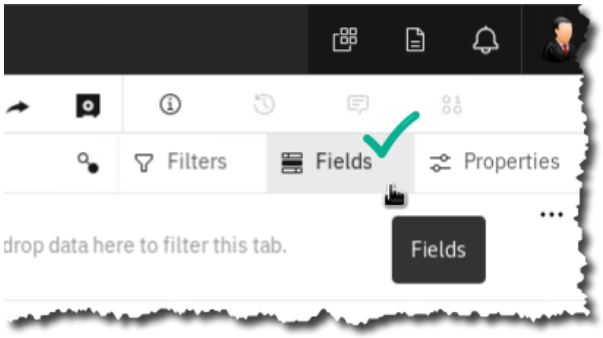
__32. Drag and drop the COMPANY column on the canvas.

(It may take a minute or two for this column to render as a widget – please be patient.)



__33. Make sure to select the COMPANY widget on the canvas.

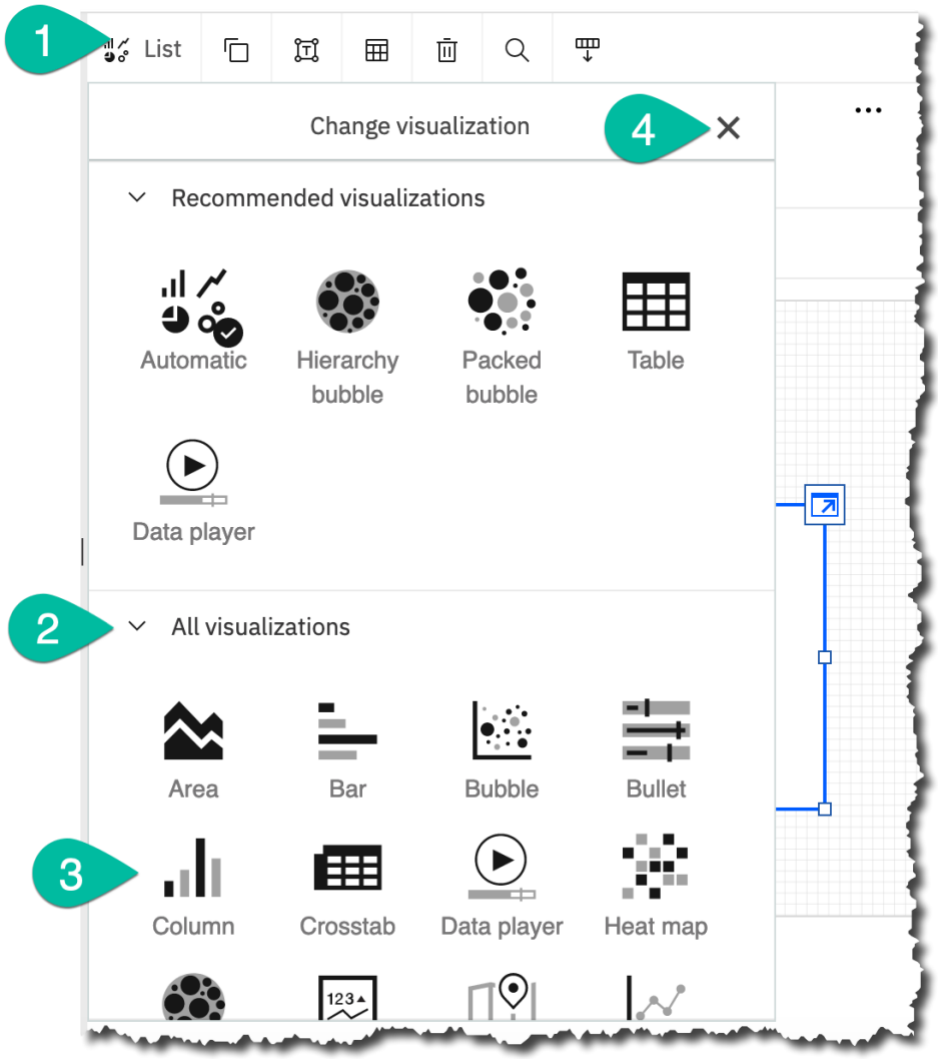
Then at the top right select the tab Fields.



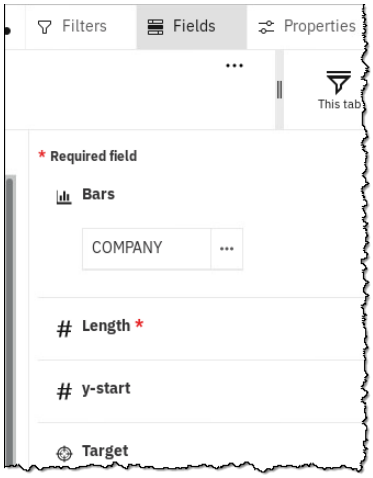
__34. On the top left (above Tab 1) change visualization type from list to column by selecting:

List box ⇨ All visualizations ⇨ Column.

Then close the visualization select box X.



__35. On the right of the screen, you should see this visualization formatting box.



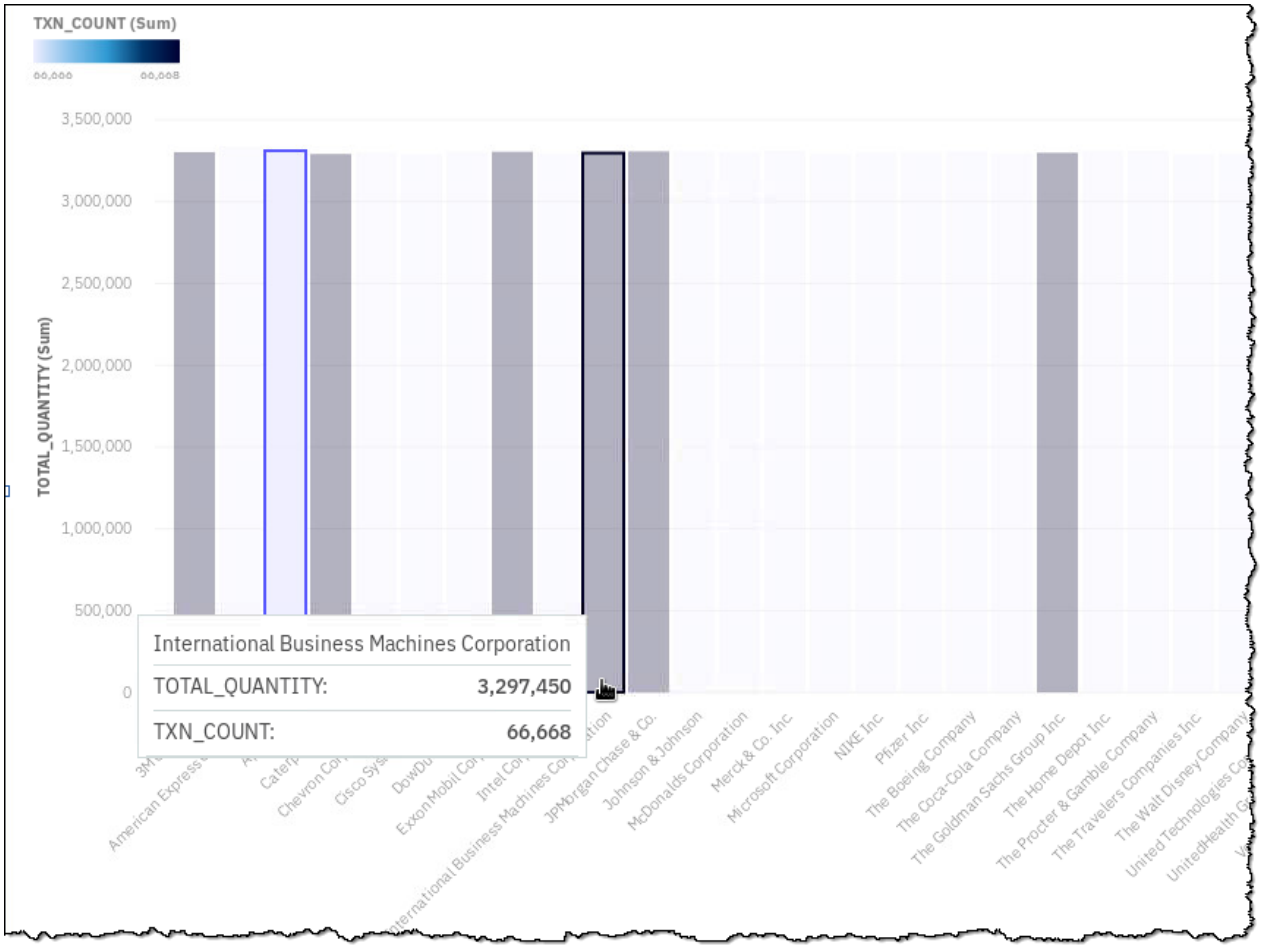
__36. Format the visualization by performing these steps, in order:

1. Make sure the COMPANY column is used for Bars (if it is not, drag it there).
2. Drag and drop the TXN_COUNT column or Color.
3. Drag and drop the TOTAL_QUANTITY column for # Length.

__37. Click the Save icon to ensure your work is saved.

__38. Perform some analysis of various companies to see the quantity traded and the transaction counts by hovering on an individual company from within the visualization.

Example: hover over the column for International Business Machines Corporation.



Business Analyst	I noticed a delay while retrieving my data in this dashboard which in the long run will be unsuitable for me to use repeatedly. I need to reach out to my CPD Administrator and Data Engineer to find a way to make this perform better.

11.7 Creating the Data Virtualization cache

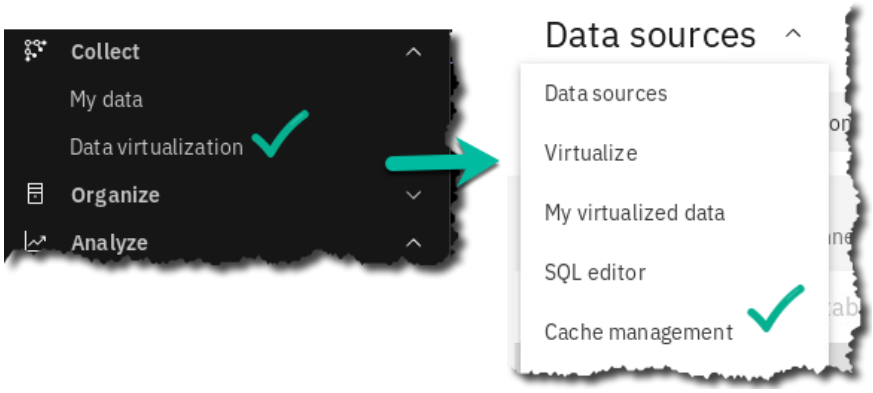
11.7.1 Creating the DV data cache

Persona (Role)
 Administrator (DV/CPD)

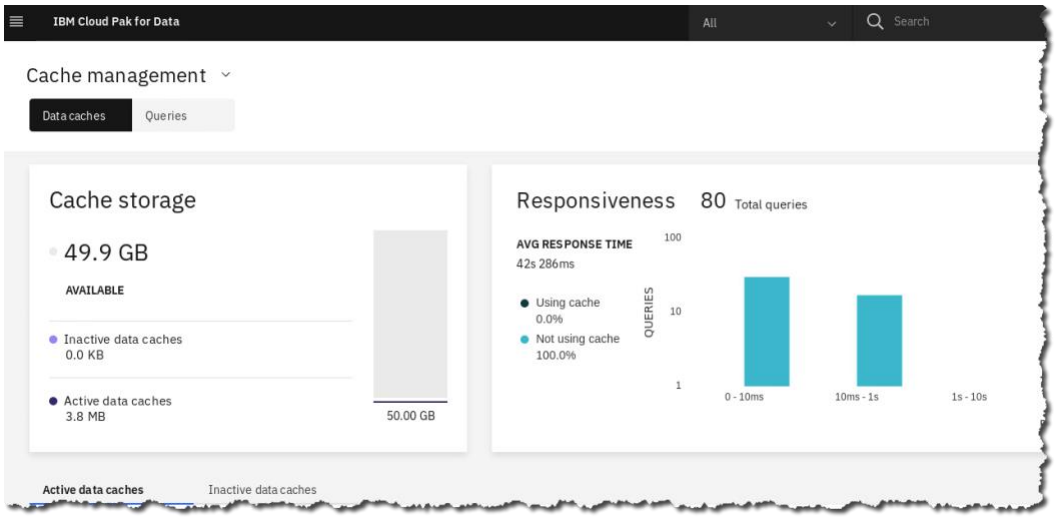
The DV Admin will first analyze the queries executed against DV and identify those generated by the dashboard. Looking at the query will help understand what type of cache needs to be created to improve performance for a better dashboard experience.

__39. Start at the Navigation Menu.

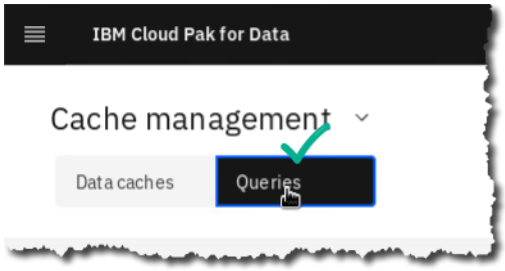
__40. Click Collect ⇨ Data virtualization ⇨ Menu (Data Sources) ⇨ Cache Management.



__41. You will be presented with your Cache management dashboard.



__42. Click tab Queries.

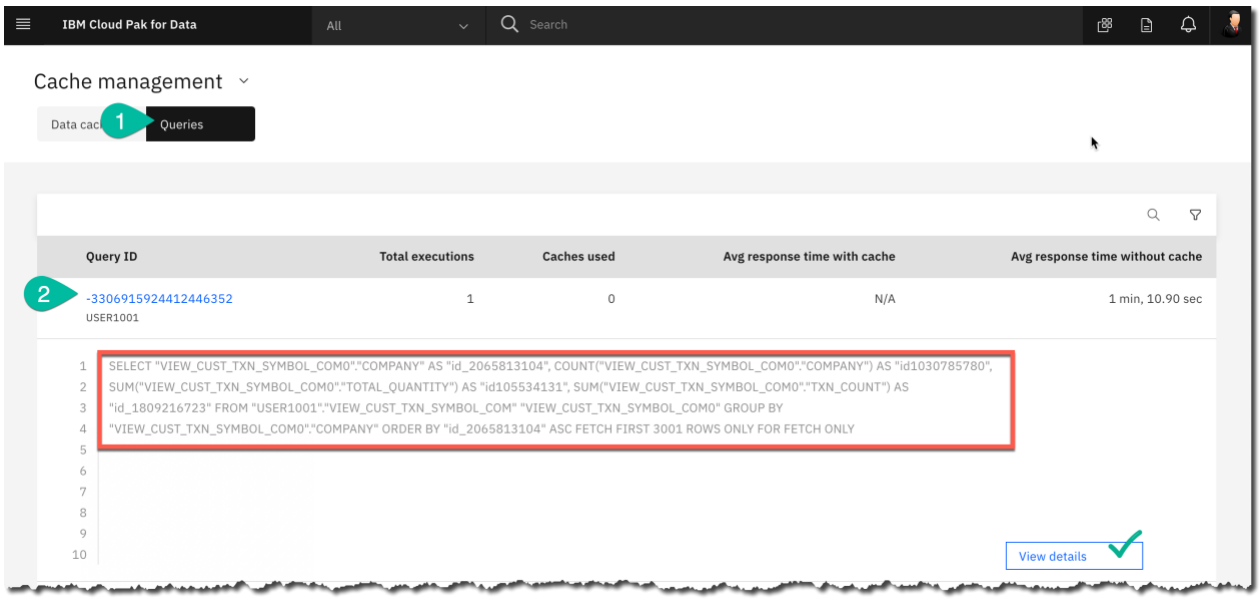


__43. At the far right of the screen select Filter ⇨ Not using cache ⇨ Response time: 1m-10m ⇨ Last 24 hours ⇨ Apply.

__44. The list of queries executed are shown with their average execution times if run more than once. Note the response time which explains the delay in working within the dashboard. (The example below is greater than 1 minute and 10 seconds)

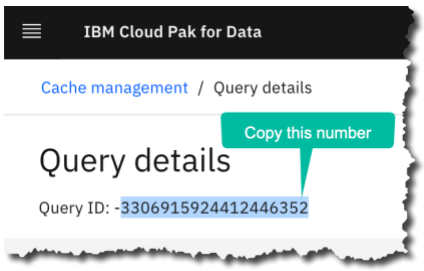
Select the down arrow by first Query ID to see the actual SQL being run.

NOTE: If the query you chose does not contain 'SELECT VIEW_CUST_TXN_SYMBOL_COM at the start of the query, use the back-arrow key to return to the cache, scroll down and view another query.



Click View details.

__45. Record (or screen shot) the Query ID, which will be used later in this lab. (Often, recording only the last 6 digits is all that is necessary.)



One could choose to create a cache from this particular query, but since the dashboards will be using all columns from the View created anyway, the DV Admin and the BA decides to create a cache for the entire virtualized view instead.

__46. Go back the main Cache Management page and click Data caches.

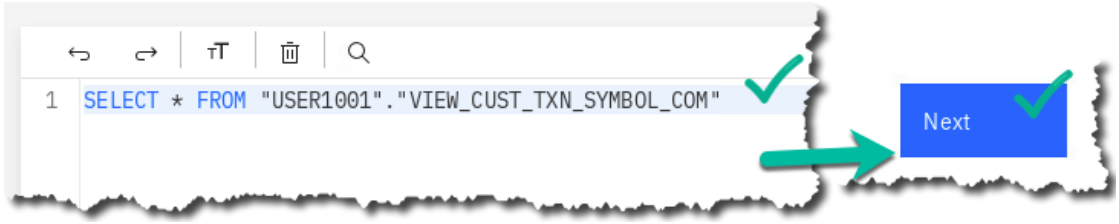
__47. At the right of the page, click Add new cache.

__48. This opens a SQL Editor to enter the query from which the cache will be created.

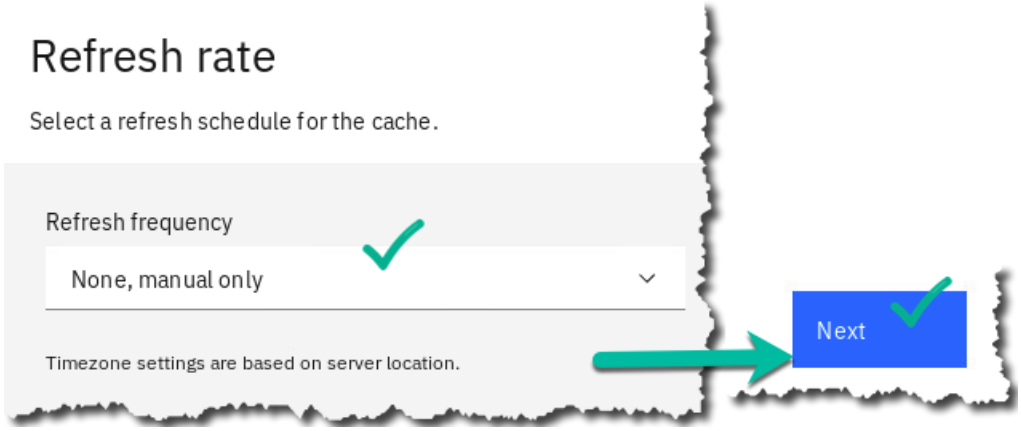
Type in the query to select all columns from the virtualized view:

SELECT * FROM "USER1001"."VIEW_CUST_TXN_SYMBOL_COM"

Click Next.



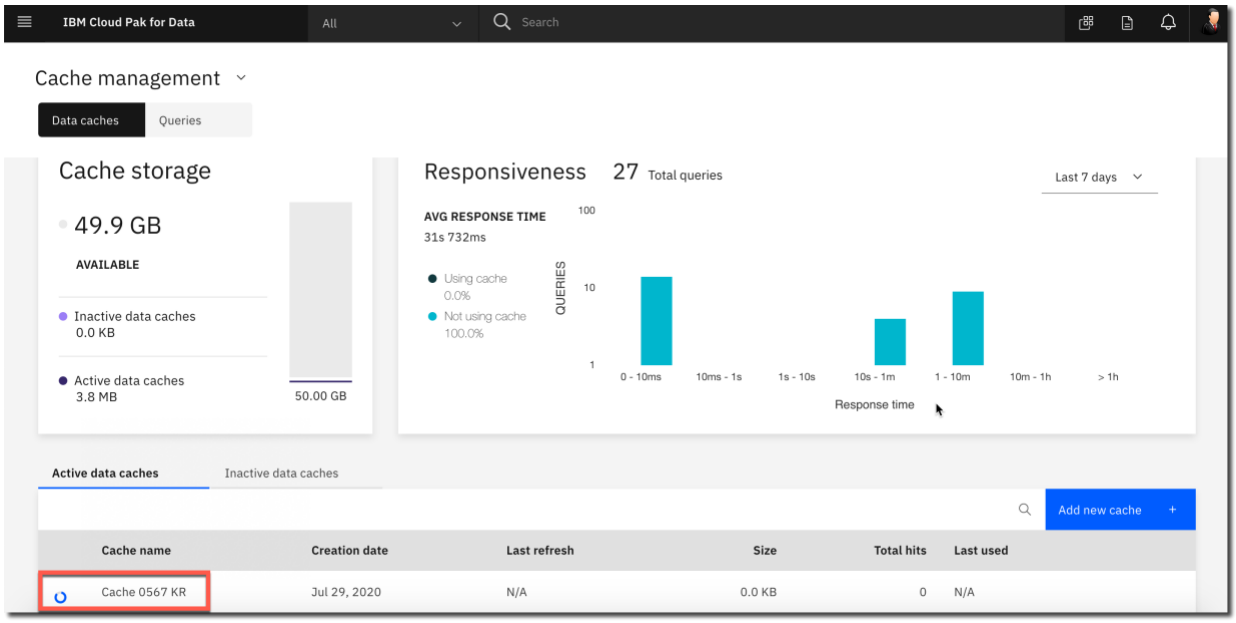
__49. In the Refresh rate choose the default (None, manual only) and click Next.



__50. Enter your initials at the end of the Cache name (yours may vary) then confirm the details on the final page.

Click Create.

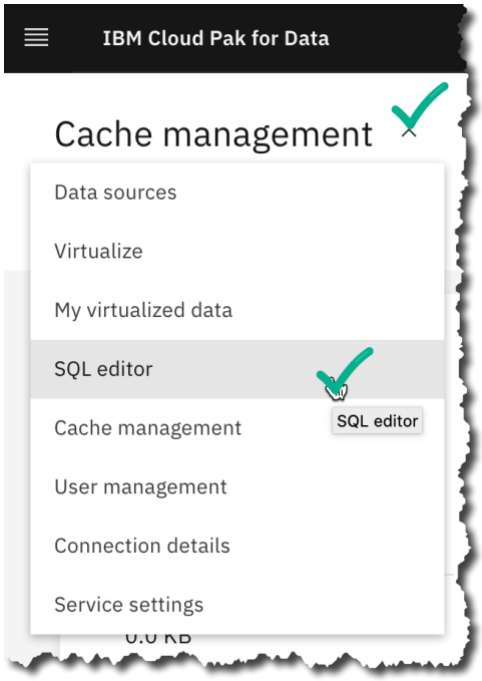
The cache creation process may take some time and the main Cache Management page will reflect the work in progress.



__51. Once the cache creation is complete, the newly created cache shows up under the Active data caches along with other details and its size.

Admin	KNOWN ISSUE – There exists a defect/issue wherein queries executed before the cache is created and activated fails to use the cache. This is because the query plan is already cached in a relational database such as Db2, so the work around is to clear the Db2 package cache and collect table statistics on the virtual tables created above. You will do that next.

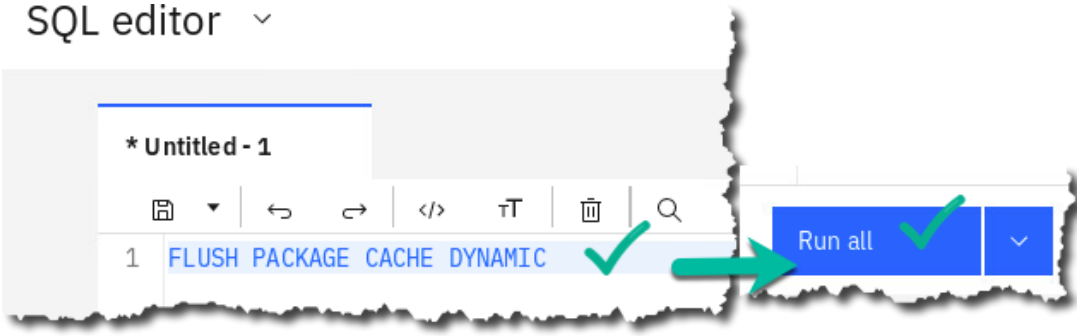
__52. Click Menu (Cache management) ⇨ SQL Editor.



__53. Remove the view SQL and replace it with the following:

FLUSH PACKAGE CACHE DYNAMIC

Click Run all.



In a production setting, flushing the entire package may not always be a good idea as it could affect query execution for all users of the database. (We did it here to make this lab simpler.)

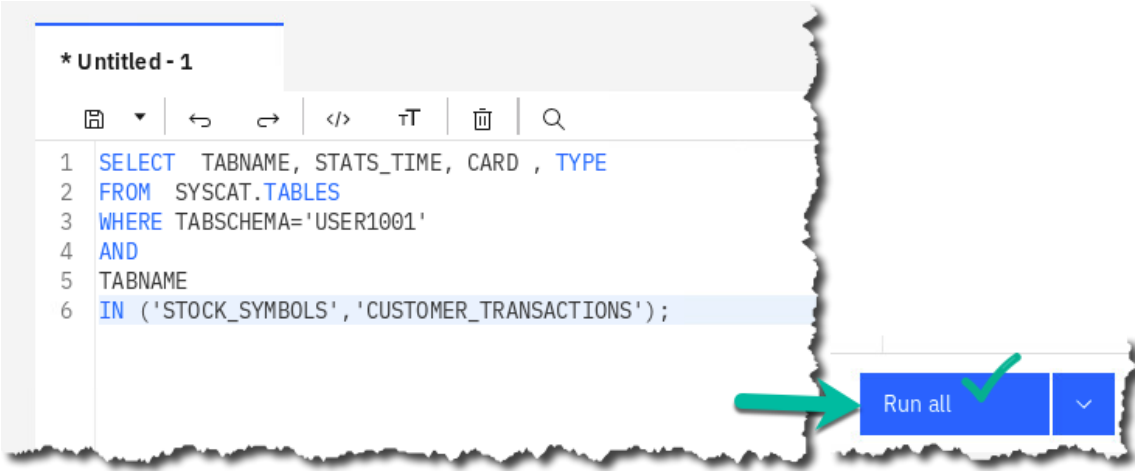
Instead, one could opt to selectively flush the package corresponding to particular queries. For more details on that see: http://ibm.biz/FLUSH-PACKAGE.

__54. Next, check if the table statistics have been collected. Replace the previous SQL with:

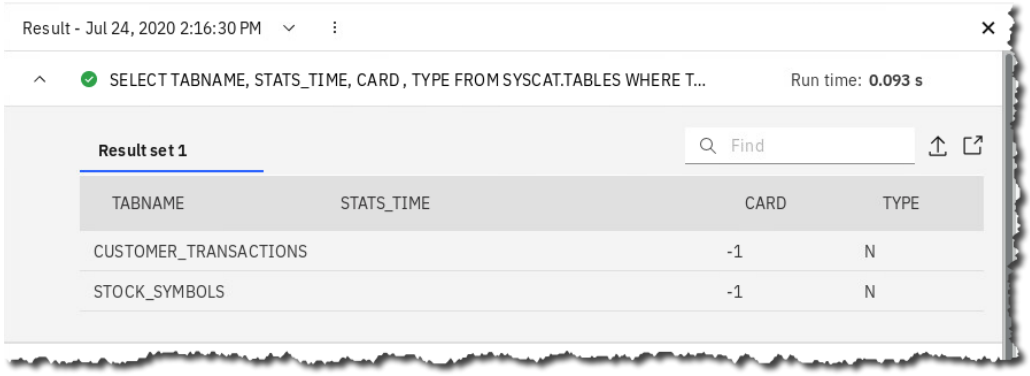
```
SELECT  TABNAME, STATS_TIME, CARD , TYPE
FROM  SYSCAT.TABLES
WHERE TABSCHEMA='USER1001'
AND
TABNAME
IN ('STOCK_SYMBOLS','CUSTOMER_TRANSACTIONS');
```

 Data Engineer	Note: You can copy and paste this SQL directly from the Unified Desktop by using the File Browser to open this file:  ⇨ Home ⇨ Downloads ⇨ DV_SelectTabname.txt. Alternately, you can download this file by opening a browser tab and using this link: https://ibm.box.com/v/DV-SelectTabname

__55. Click Run all.



If the cardinality for the virtual tables created before shows -1, this indicates that table statistics have not been collected. These statistics will be collected now.



 Data Engineer	Note: The SYSPROC.NNSTATS is a procedure to collect statistics for remote tables: http://ibm.biz/SYSPROC-NNSTATS.

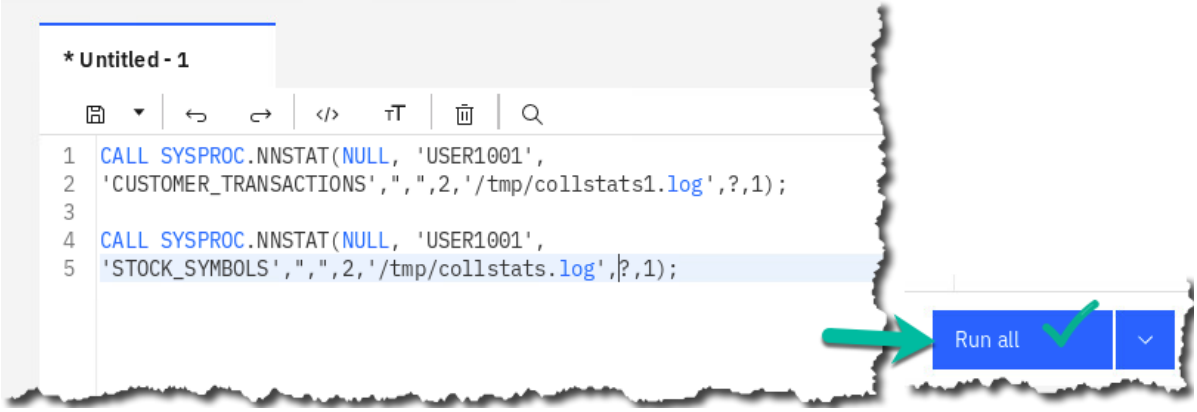
__56. If your statistics are not current, then do this step. If they are current, you can skip this step.

CALL SYSPROC.NNSTAT(NULL, 'USER1001', 'CUSTOMER_TRANSACTIONS','', '',2,'/tmp/collstats1.log',?,1);

CALL SYSPROC.NNSTAT(NULL, 'USER1001', 'STOCK_SYMBOLS','', '',2,'/tmp/collstats1.log',?,1);

 Data Engineer	Note: You can copy and paste this SQL directly from the Unified Desktop by using the File Browser to open this file:  ⇨ Home ⇨ Downloads ⇨ DV_CallSysProc.txt. Alternately, you can download this file by opening a browser tab and using this link: https://ibm.box.com/v/DV-CallSysProc .

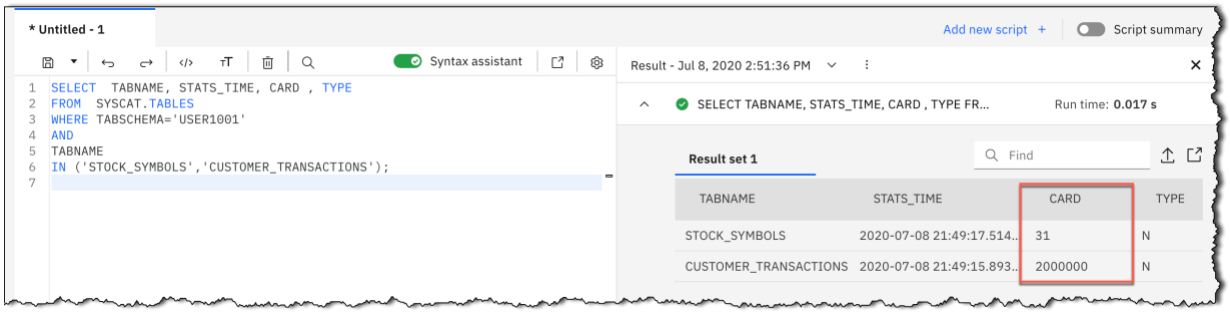
__57. Click Run all.



__58. To confirm if the table statistics have now been collected, replace the previous SQL with the following:

```
SELECT  TABNAME, STATS_TIME, CARD , TYPE
FROM  SYSCAT.TABLES
WHERE TABSCHEMA='USER1001'
AND
TABNAME
IN ('STOCK_SYMBOLS','CUSTOMER_TRANSACTIONS');
```

__59. Click Run all.



The actual rows counts for the virtual tables are shown as expected instead of -1.

With the package cache flushed and the table level statistics collected, all queries henceforth referencing the virtualized view should start using the cached copy instead of accessing the underlying data sources directly.

The DV Admin informs the BA about the cache creation and asks the BA to re-run the dashboard.

Persona (Role)
 Business Analyst

__60. Return to the dashboard created previously.

Navigation Menu ⇨ Projects ⇨ your_initials-DV-Project ⇨ Assets ⇨ Dashboards ⇨ Company Stock Analysis

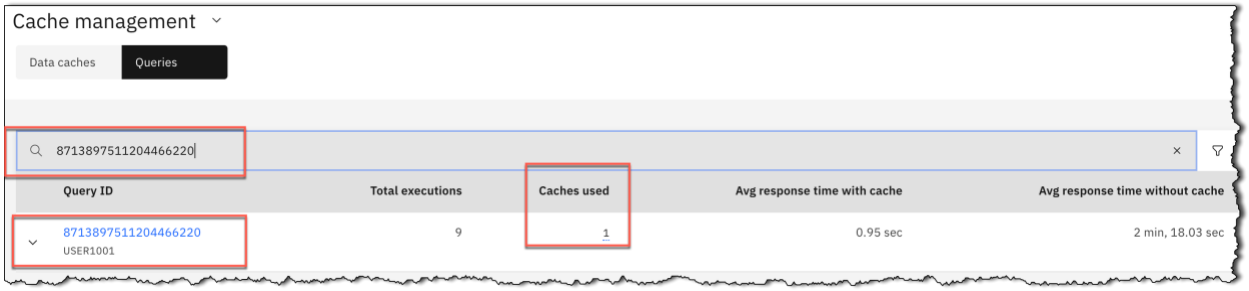
__61. The dashboard should now render faster with much better response times while analyzing the stock data. The queries executed by the dashboard against the virtualized view now use the DV cache.

__62. Return to Data Virtualization, to Menu ⇨ Cache Management ⇨ Queries to get some insights into the queries you just ran.

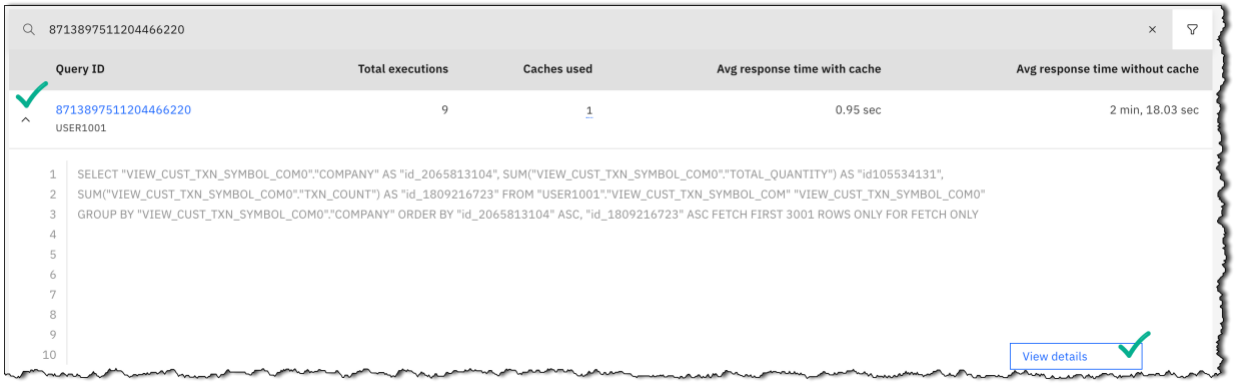
Note: Because there is sometimes a slight delay in getting all the information about the latest queries and cache info via the UI, you can search for the query by Query ID (that you hopefully noted in the previous step.)

Click on the Search (magnifying glass) icon.

Enter the Query ID to list the query.



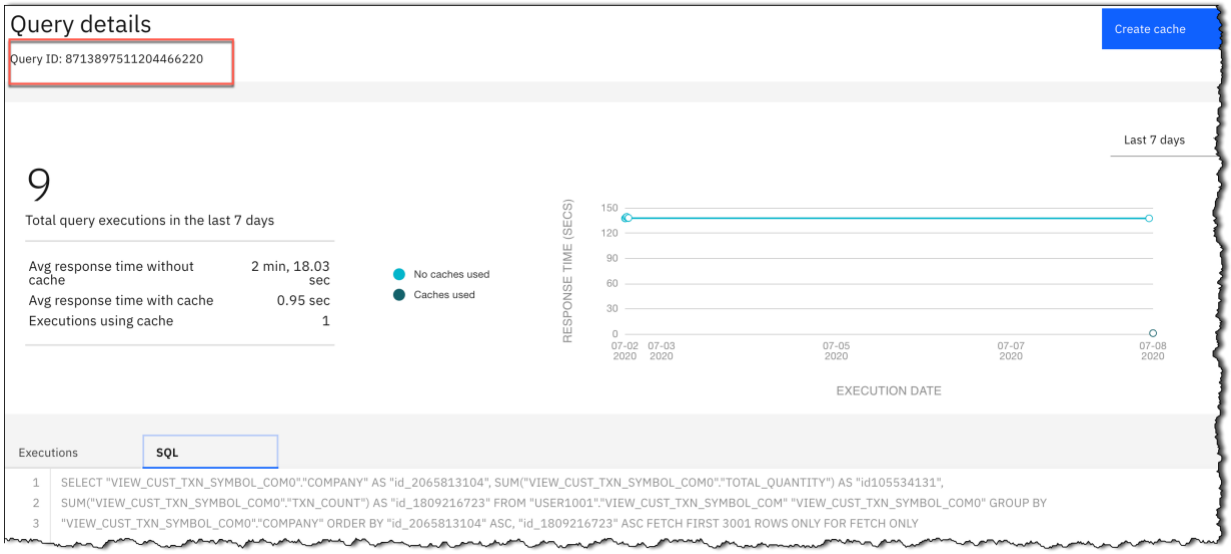
__63. Expand the entry to view the query details.



__64. Click on View details to show the number of executions and execution times.

Notice in the above example (yours may vary somewhat) the query took under 1 second to run with the cache, while without cache it was 2 minutes and 18 seconds.

__65. Confirm that this is the same query that was executed in the previous steps, by confirming the Query ID and by clicking on SQL.

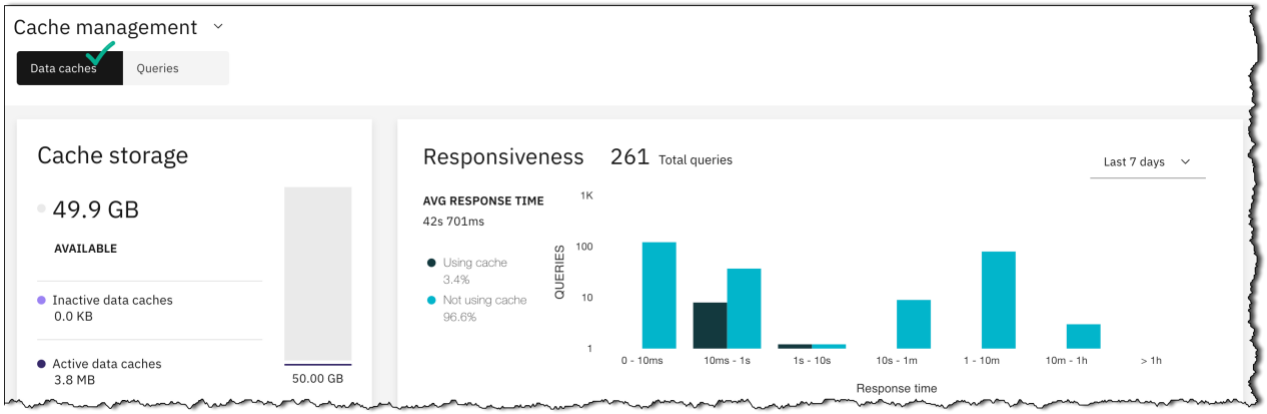


The query execution time has reduced **significantly** with the cache now being utilized instead of fetching the data from the underlying sources!

The cache hits should also start showing up under Cache Management ⇨ Data caches.

11.8 Reviewing the Cache Management UI

The Cache Management UI helps the DV Admin to create, monitor and manage caches. It provides information about the active/inactive caches like size, hit counts etc. along with options to work with them.

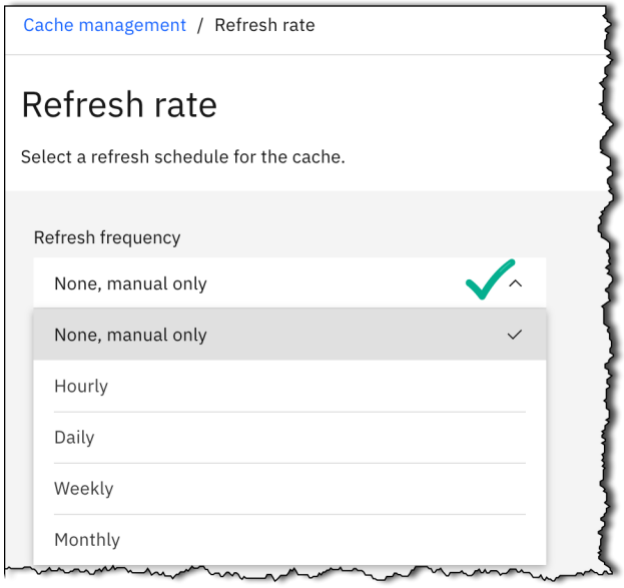


The UI also provides details about the queries executed against DV. This is particularly useful for the DV Admin in identifying slow running queries and potentially creating caches to make them run faster.

Play with some of the features of the Query section using Filters.

11.9 Refreshing the Data Virtualization cache

As data changes in the underlying data sources, it is important to refresh the cache and reflect the freshly updated data. The cache creation process provides an option to set the cache refresh rate.



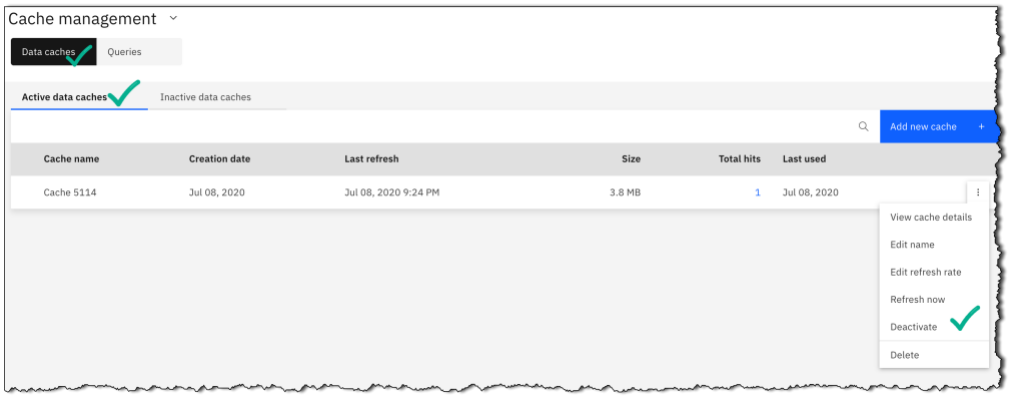
One can choose from the following options in the drop down:

1. *None, manual only:* After the initial refresh, all subsequent refresh has to be manually performed by the DV Admin.
2. *Hourly:* The cache gets refreshed hourly on the set minute of the hour.
3. *Daily:* The cache gets refreshed at a specified time of day.
4. *Weekly:* The cache gets refreshed at a specified time on specific day(s) in a week.
5. *Monthly:* The cache gets refreshed at the specified time on specific day(s) and week(s) in a month.

To avoid delays and unexpected results, the best time to refresh the cache should be chosen based on the lowest DV usage/low query traffic.

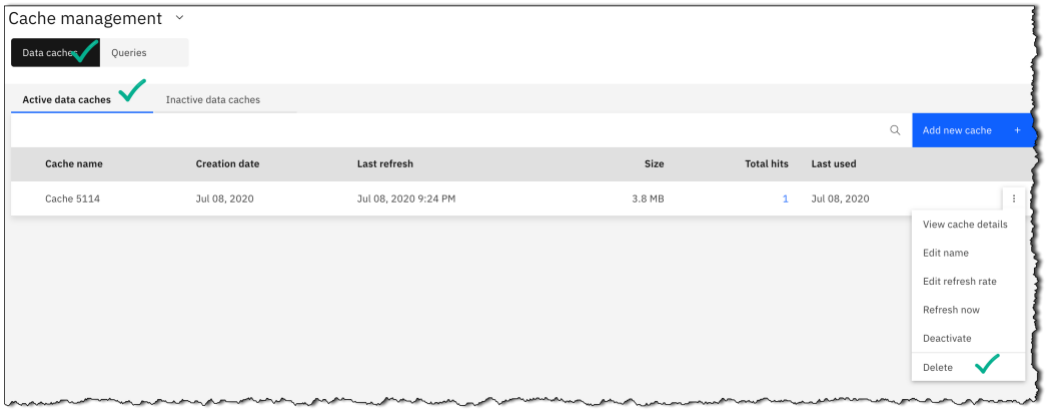
11.10 Deactivating/Activating/Deleting the Data Virtualization cache

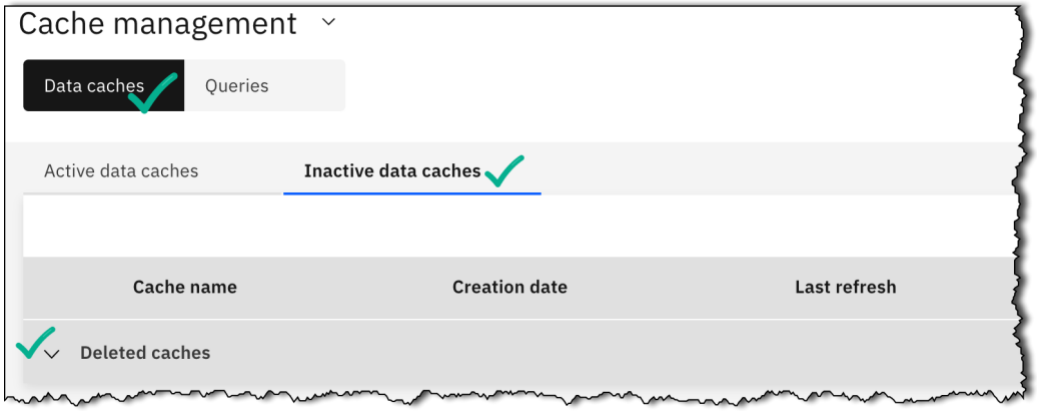
The Cache Management UI provides some options to help manage caches. After a cache is created, it gets refreshed/populated by data and remains in the active state by default, to be considered by the Query Optimizer while generating the query plans.



An active cache can be deactivated so the Query Optimizer will not consider it while generating the query plans. This could be useful if there are issues related to the cached data and needs investigation.

A cache can be deleted, which moves it into the inactive state and eventually cleaned up. The list of deleted caches can be listed using the Deleted caches option. It shows the original cache details and if needed, you could re-create them.





11.11 Caching guidelines

The DV cache is a very useful feature for improving DV query performance. However, there are some important points to remember:

1. The DV cache can only be created by the DV Admins.
2. A cached object has the same permissions as the underlying DV object.
3. Use queries that perform some sort of aggregation from the underlying data to create the cache. Performing aggregations help create summaries, and thus limit data cached.
4. When possible avoid duplicating data that exists in its original form in the underlying data sources.
5. Identify the most commonly used query or parts of queries and consider creating the cache from them. That way multiple users and queries can benefit from those caches.
6. Limit the size of the data to be cached. Note that the cache uses the storage allocated to the DV instance, hence it is common to all DV users. Large caches will limit the availability of space for other users/scenarios.
7. Time taken to refresh a cache will depend on the amount of data, latency and availability of the underlying data sources.
8. A cache will not be used for a query if there is no match for the statement or column(s) and the virtual object being queried.
9. The DV Cache Management UI takes couple of minutes to refresh the queries executed, cache hit counts and other relevant information.

11.12 Lab conclusion

Data Virtualization (DV), as part of the Collect phase, facilitates accessing data from various data sources and running queries across them. Since data movement is limited, all of the access rules and policies for creating copies remain preserved. In situations where query response times are paramount, DV provides the caching facility, which was covered in this lab.

**** End of Lab 11 - Collect: Data Virtualization Caching – Deeper Dive**

Lab by Rajesh Kartha, Edited by Burt Vialpando and Kent Rubin

Lab 12 COLLECT: VIRTUALIZING & CACHING FROM Z/OS – DEEPER DIVE

12.1 Lab overview

Data Virtualization (DV) is an important component of IBM Cloud Pak for Data (CPD) to help integrate data sources across multiple types and locations and turn them into one logical data view. As shown in the previous DV Caching Deep Dive lab, virtualizing tables from databases hosted in the cloud and then leveraging caching helps to drastically improve query performance.

In addition to the cloud, almost all enterprises will have large amounts of data stored in on-premises data sources. Data hosted on z/OS systems can be a major portion of that.

While there exist multiple ways to access data residing in a z/OS system, using IBM Data Virtualization Manager (DVM) is one of the most popular ones. Cloud Pak for Data Virtualization (CPD-DV) has the ability to virtualize and ingest any mainframe data that is available, leveraging the Data Virtualization Manager for z/OS.

12.2 IBM DVM

IBM Data Virtualization Manager (DVM) for z/OS® provides virtual, integrated views of data residing on IBM Z®. It enables users and applications read/write access to IBM Z data in place, without having to move, replicate or transform the data.

IBM DVM enables data structures that were designed independently to be used together. Traditional data movement approaches can negatively impact the opportunity to benefit from data where and when it is needed. By unlocking IBM Z data using popular, industry-standard APIs, DVM for z/OS can save you time and money.

DVM for z/OS provides access to IBM Z data sources such as VSAM (Virtual Sequential Access Method), IBM Db2 for z/OS, Adaptable Database System (ADABAS), Integrated Database Management System (IDMS), IBM Information Management System (IMS), IBM System Management Facilities (SMF) and non-IBM Z data sources, all without the need for mainframe skills.

You can also simplify the development of AI applications directly from IBM Z data with CPD.

For more information on DVM for z/OS, check out this link: http://ibm.biz/DV-Manager-ZOS.

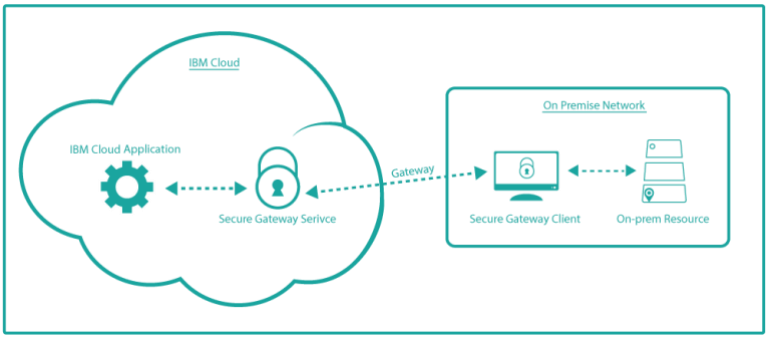
Also check out the IBM Demos page: http://ibm.biz/IBM-Demos

12.2.1 Secure Gateway

Accessing on-prem data from an external system can be challenging and there are different ways to achieve that. A simple and cost-effective way in the IBM Cloud is to use the Secure Gateway service.

IBM Secure Gateway for IBM Cloud maintains a single persistent encrypted connection between the Secure Gateway client in the on-prem network and the Secure Gateway server in the IBM Cloud. Data can be securely transmitted bidirectionally between on-prem and external resources.

The IBM Secure Gateway (SG) consists of a Gateway service, an on-prem SG Client and an on-prem Destination.



External applications can securely connect to the Gateway service, which in turn connects to the on-prem destination via the configured SG client. The external applications are unaware of the on-prem destination details. Theyonly knows the IBM Cloud Secure Gateway service endpoint and the corresponding credentials to be used. More information about IBM Cloud Secure Gateway service is available in its documentation: http://ibm.biz/Getting-Started and http://ibm.biz/Client-Install.

In this lab, you will create a DV virtualized table that spans three different data sources: one z/OS on-prem data source via IBM DVM, and two Db2 Warehouse tables hosted in the IBM Cloud. This allows querying across all of them via a single DV query and then uses the caching capability of DV to improve query times.

In our scenario, the Trade Co. Business Analyst (BA) builds on the dashboard created in the previous DV Caching Deeper Dive lab using data from a DV view, which in turn points to two remote IBM Cloud Db2 Warehouse data sources underneath. However, there is now a need for more historical analysis with the addition of a third z/OS data source that hosts transaction data from previous years. The BA works with the Data Engineer and DV Admin, who can create the necessary virtual objects (tables and views) and a data cache for the final virtualized view to help address the data retrieval performance issues.

12.2.2 Setup (informational only)

Information in this section is only for information purposes. All the setup steps have already been performed on your workshop cluster to facilitate this lab completion.

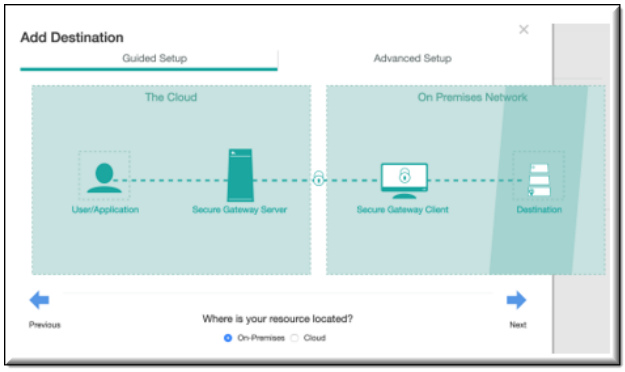
z/OS Data:

In this lab example, a transactional process on z/OS has stored records of stock market trades into a VSAM data set. VSAM data sets arrange data in fixed format records by an index key, by a relative byte address, or by a relative record number. On the mainframe, VSAM data sets are cataloged and used frequently for easy retrieval by z/OS application and processes. DVM for z/OS has the ability to create a customized virtual table or virtual view definition over this VSAM data set.

CPD-DV, using a DVM for z/OS connection, can query the virtual table defined over the VSAM data using ANSI-standard SQL, and virtualize this stock trade data with other DV data sources.

Access:

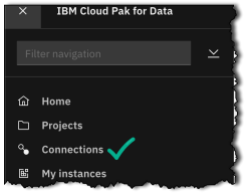
The VSAM data and the DVM for z/OS application reside on a mainframe server (z/OS) at the Washington Systems Center on IBM premises. This system is in a secured network zone and access from the CPD system is provided through a Secure Gateway client, running inside the IBM network configured to an IBM Cloud Secure Gateway service.



Connection:

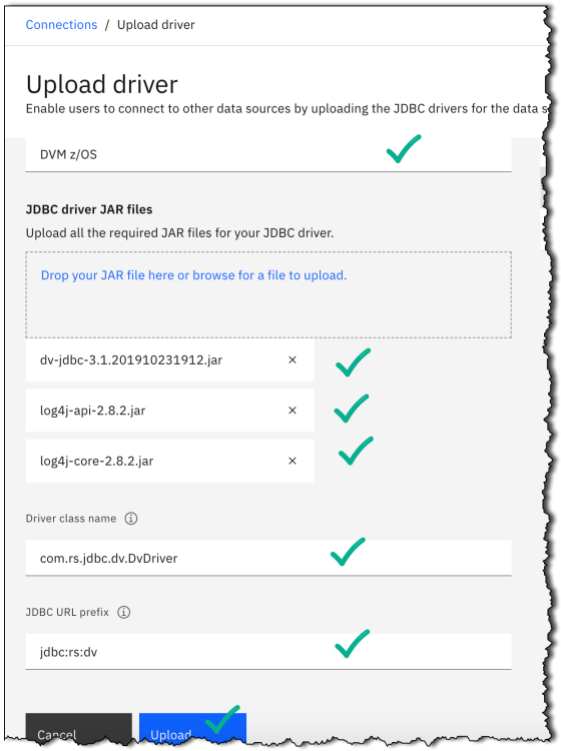
IBM DVM for z/OS comes with its own set of JDBC drivers and CPD-DV uses these to connect and virtualize tables from the z/OS system. For this lab, these JDBC drivers have already been loaded into the Cloud Pak for Data instance and are ready to create a connection. We will show you how we did it here. ***NOTE: You do not need to complete these steps for this workshop.***

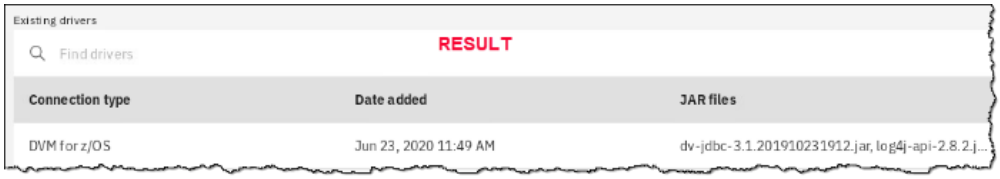
To upload the DVM JDBC driver, from the navigation menu go to Connections.



Click on Upload driver.

Provide a name and upload the JDBC jars along with the names of the Driver class name and the JDBC URL prefix. Click on Upload.





12.3 Personas represented in this lab

The Business Analyst persona along with the Data Engineer and Data Virtualization Administrator personas will perform the exercises in this lab.

Persona (Role)	Capabilities
 Business Analyst	Business Analysts deliver value by taking data, using it to answer questions, and communicating the results to help make better business decisions.
 Data Engineer	Data Engineers build and optimize the systems to allow data scientists and business analysts to perform their work. The Data Engineer ensures that any data is properly received, transformed, stored, and made accessible to other users.
 Administrator (DV/CPD)	Administrators set up and maintain the DV module within the CPD environment itself. They are responsible for granting DV access to users and administration tasks like creating a data cache.

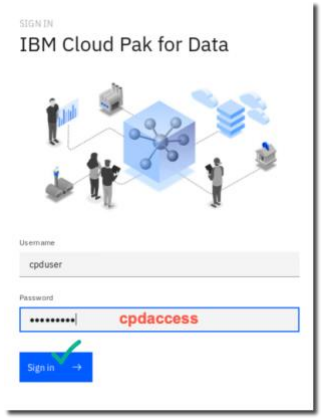
For convenience in doing this lab, instead of switching between personas, all the required privileges have been provided to the same user. The workbook will refer to the respective personas at different stages to help understand the flow of this task.

12.4 Logging into the CPD web client (if you have not already done so)

__1. If you are starting this lab stand-alone (without going through previous labs) do the following:

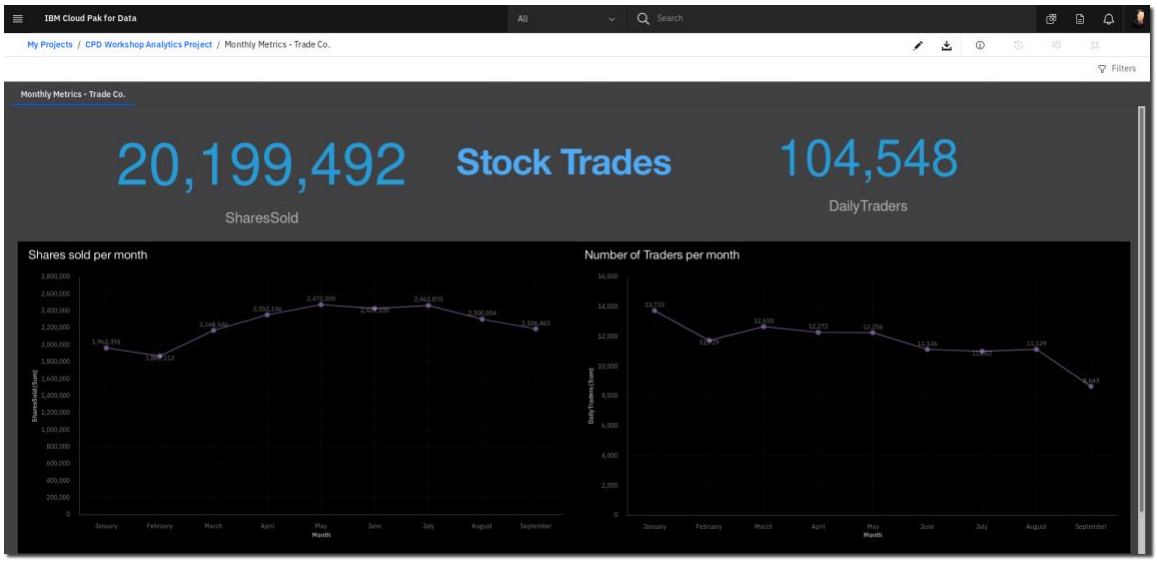
__2. Double-click the desktop icon: Cloud Pak for Data Web Client.

__3. The CPD web client GUI displays as shown. Use cpduser and cpdaccess for the *Username* and *Password* and click Sign in.



12.5 Reviewing the dashboard: Stock Trading Analysis - Trade Co.

In the attempt to understand stock trading patterns, the business analyst for Trade Co. starts by creating a simple dashboard to find the most popularly traded stocks historically. The dashboard shows the number of Shares Sold vs. the number of Daily Trades.



This lab follows the same narrative as that of the DV Caching Deeper Dive lab. The Business Analyst (BA) works with the Data Engineer (DE) to get virtualized access to the different data sources required for creating the dashboard(s). The DE creates the Data Sources, Virtual Tables and finally a Virtualized View joining all the Virtual Tables and computing the basic aggregations required. The View is then shared with the BA, who can then proceed with creating the dashboard.

However, once the dashboard is initially created, the BA notices delays in rendering the visualizations. Since every request from the dashboard has to fetch the data from its original source(s), latency starts to play an important role, slowing down response. The BA works with the DV Admin to create a cache for the View, which helps speed up query times significantly and hence rendering of the dashboard(s).

12.6 Adding the DV connection to DVM for z/OS

The Data Virtualization process begins by adding data sources to virtualize and is typically done by the Data Engineer.

Persona (Role)

Data Engineer (DV/CPD)

__4. Before proceeding with the creation of the data connection, ensure that the data source is accessible.

Click on this embedded link to confirm, or type: https://ibm.biz/cpd-workshop-dvm.

The above link (http://dvmtestapp4dv.us-south.cf.appdomain.cloud/TestCP4DDVM) is a verification test running in the IBM Cloud to confirm if the z/OS and DVM systems are accessible and hence able to be queried.

 If the above link times out or throws an error like this:

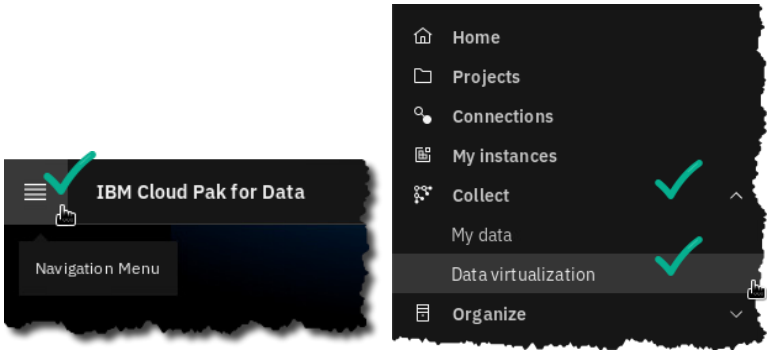
that means that the z/OS and DVM systems are not accessible for some reason.

Please contact Rajesh Kartha. Email kartha@us.ibm.com Slack: @Rajesh Kartha.

12.6.1 Navigating to Data virtualization

__5. Start at the Navigation Menu.

Click Collect ⇨ Data virtualization.



12.6.2 Creating a new data connection to DVM for z/OS

__6. The JDBC drivers for connecting to DVM for z/OS have already been loaded and the system is ready to create a connection.

Click Add new data source.

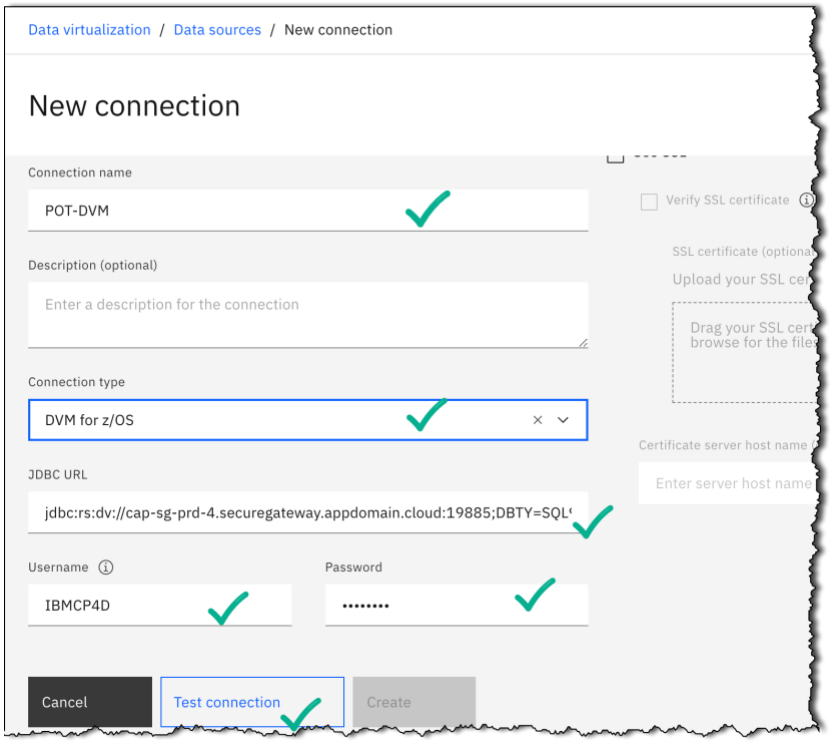
__7. Enter the details from the table below and select DVM for z/OS from the drop down.

Note: You can get the Connection information by downloading this file: http://ibm.biz/DV-Z-OS.

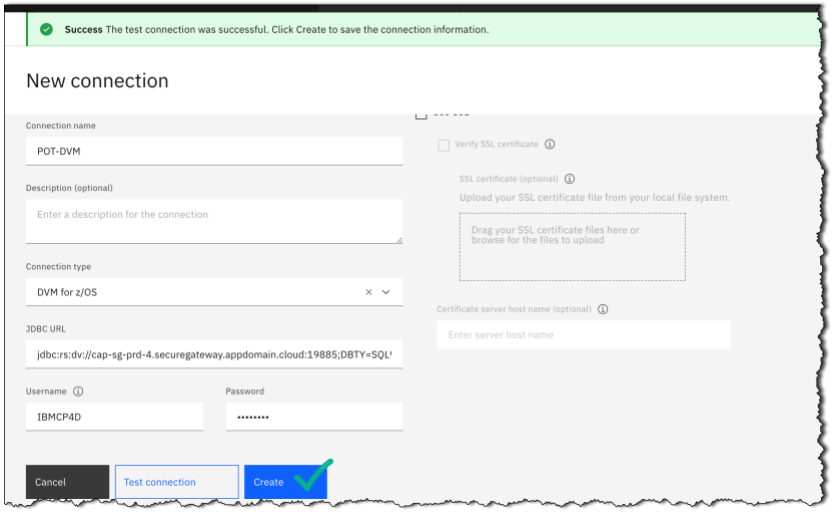
(If you are using the Unified Desktop, simply open a browser and type in the URL.)

Connection Name	POT-DVM
Description	DVM Connection
Connection type	DVM for z/OS
JDBC URL	jdbc:rs:dv://cap-sg-prd-4.securegateway.appdomain.cloud:19885;DBTY=SQL92;PMDSQL=false
Username	IBMCP4D
Password	CP4D2020

__8. Click on Test connection to test it.



__9. Once the Test connection succeeds, click on Create to create the connection.



The connection is successfully created.

12.7 Remove existing virtual tables and views (if they exist)

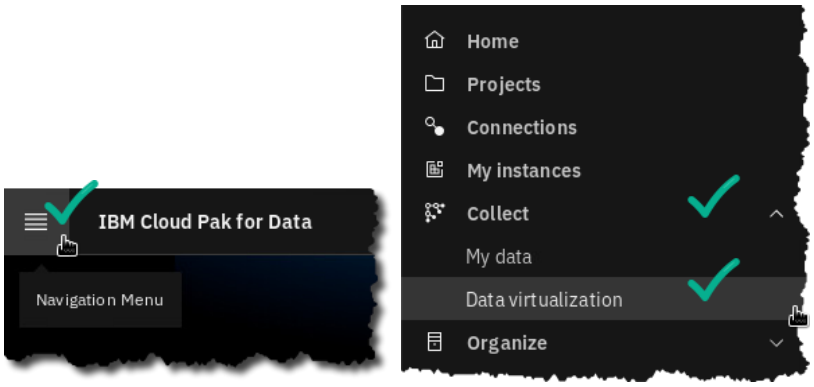
The Data Virtualization process begins by adding data sources to virtualize and is typically done by the Data Engineer.

Persona (Role)
 Data Engineer (DV/CPD)

12.7.1 Navigate to Data virtualization

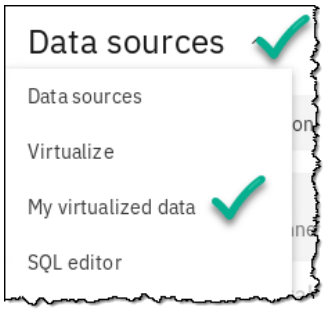
__10. Start at the Navigation Menu.

Click Collect ⇨ Data virtualization.



12.7.2 Delete virtual views and tables from the previous DV Caching lab

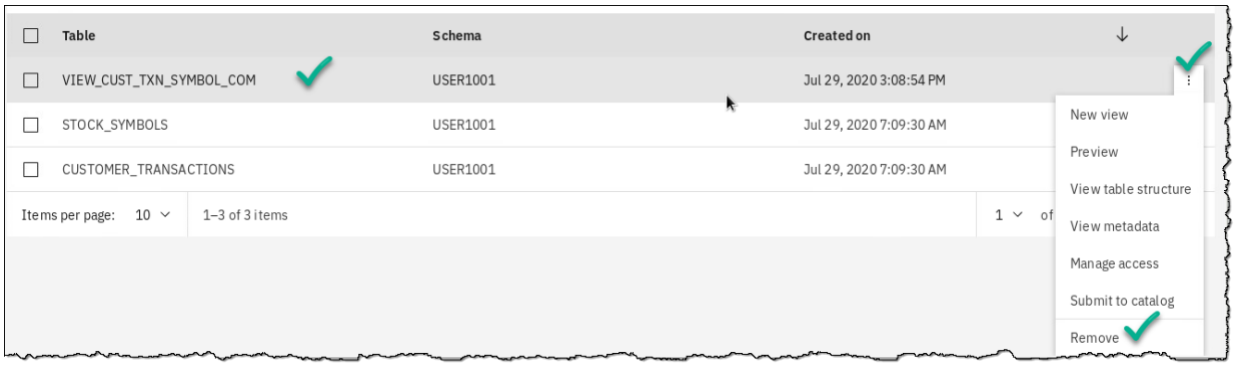
__11. From the drop-down menu (Data Sources) select My virtualized data.



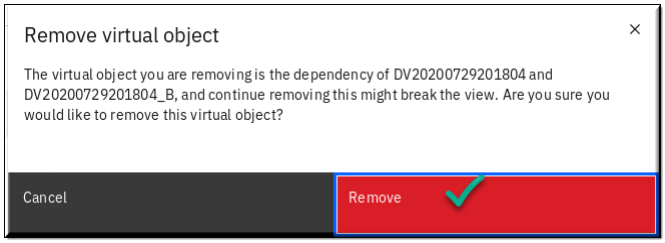
__12. Find the virtualized table VIEW_CUST_TXN_SYMBOL_COM.

(Note: If you did not do the Data Virtualization Deeper Dive lab before this one, then these assets will not exist and you can skip this and go to the next section.)

Click the ellipses on that table (at the end of the line) then Remove.



__13. Click Remove to confirm.



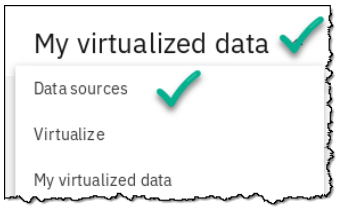
__14. In the same way, remove virtual tables: STOCK_SYMBOLS and CUSTOMER_TRANSACTIONS.

12.8 Virtualizing the remote tables and creating a view

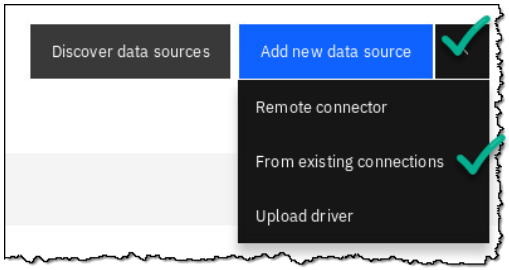
12.8.1 Add the DVM data source

__15. With the connection added in the section Adding the DV connection to DVM for z/OS, we will now add that as a data source to our Data Virtualization environment.

From the drop-down Menu (My virtualized data) select Data sources.



__16. On the top far right, select Add new data source ⇨ From existing connections



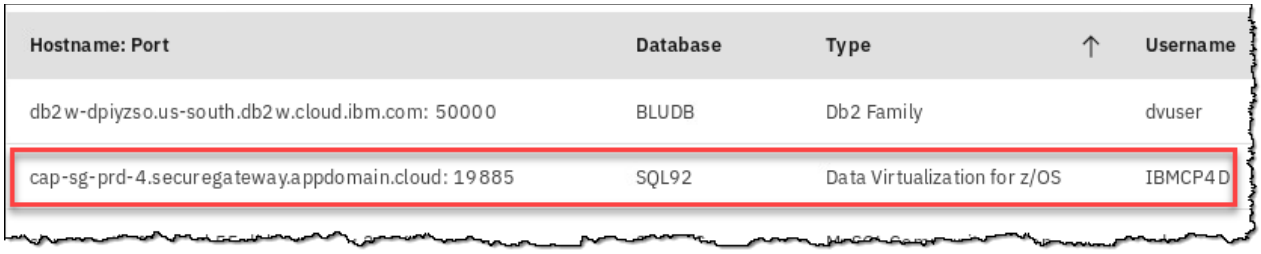
__17. Select the button for POT-DVM ⇨Next.


Data Engineer

If you receive an error message that says the data source is already created, that is OK.

Just move on to the next step. The data source should be available to review. Click the refresh button on your browser to make sure you can see it.

__18. You should now see this data source in your DV Data Sources screen.

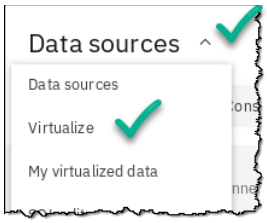


Hostname: Port	Database	Type	Username
db2w-dpiyzso.us-south.db2w.cloud.ibm.com: 50000	BLUDB	Db2 Family	dvuser
cap-sg-prd-4.securegateway.appdomain.cloud: 19885	SQL92	Data Virtualization for z/OS	IBMCP4D

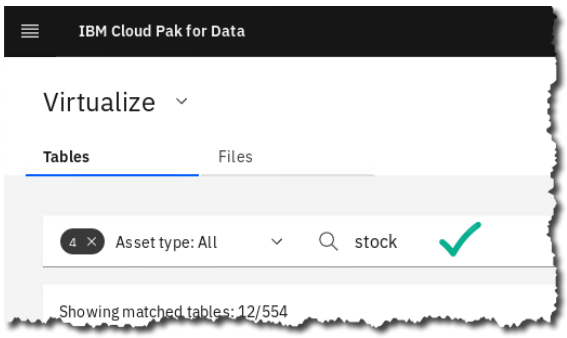
12.8.2 Create virtualized tables

__19. With the data sources successfully created, the next step is to virtualize the tables needed for this exercise.

From the drop-down menu (My Data sources) select ⇨ Virtualize.

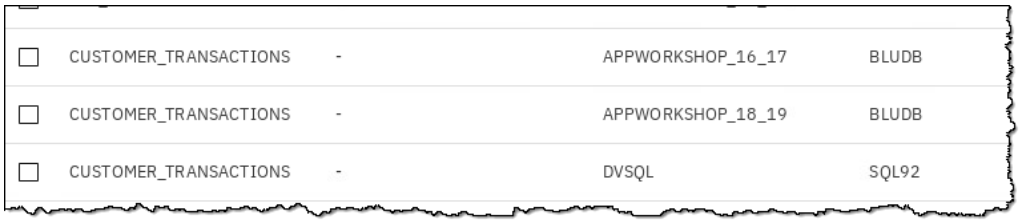


__20. In the search bar, enter the string stock.

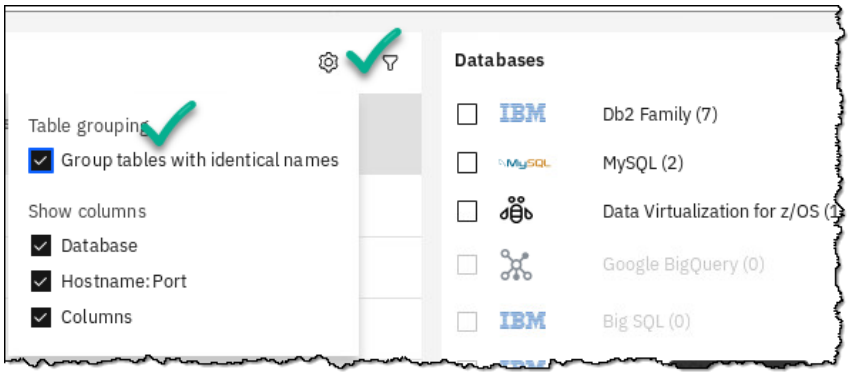


__21. Notice there are three tables available called CUSTOMER_TRANSACTIONS.

The first two tables are from two Db2 Warehouse on Cloud databases called BLUDB, and the third table is from our z/OS database called SQL92 that is available through the DVM connection you created earlier.



__22. Click the gear icon, then select Group tables with identical names.

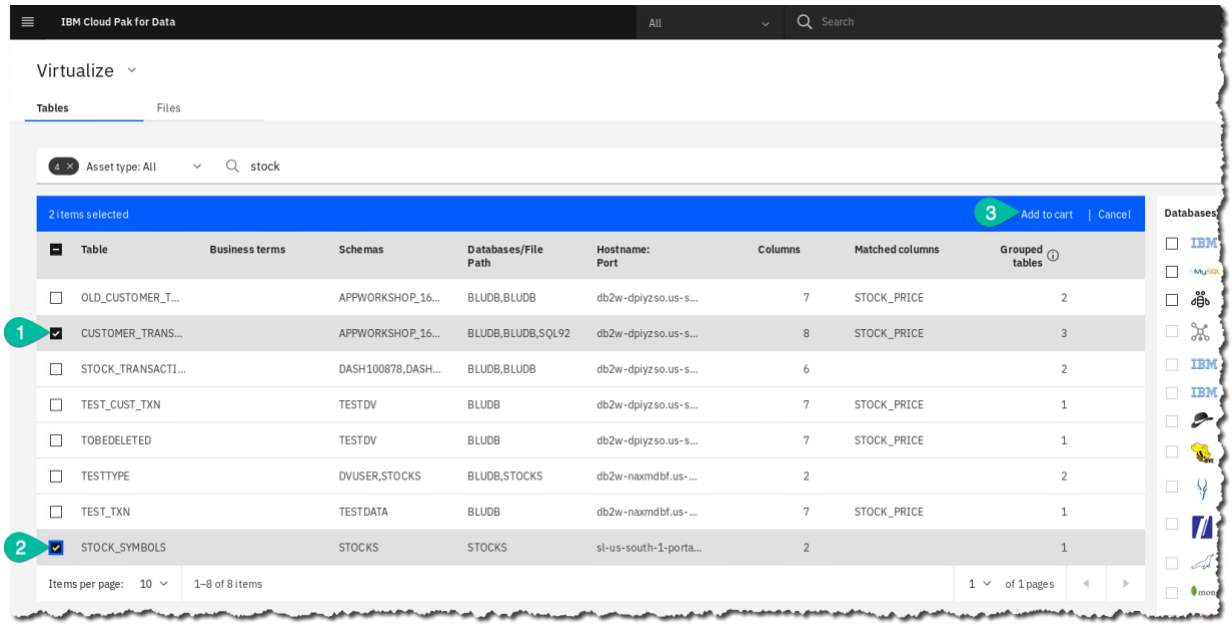


__23. This “Table Grouping” (AKA “Schema Folding”) leverages a powerful DV feature that groups tables with the same names across data sources and presents them as a single entity. For you RDBMS people, think of it as a “union all” join across databases of different types (Db2, SQL Server, Oracle, Db2 z/OS, etc.) located anywhere in the world, all with one click of a button!

Example: Notice how three tables called CUSTOMER_TRANSACTIONS from three different databases (two called BLUDB, one called SQL92) are treated as one table.

__24. Select tables CUSTOMER_TRANSACTIONS and STOCK_SYMBOLS.

Click Add to cart.



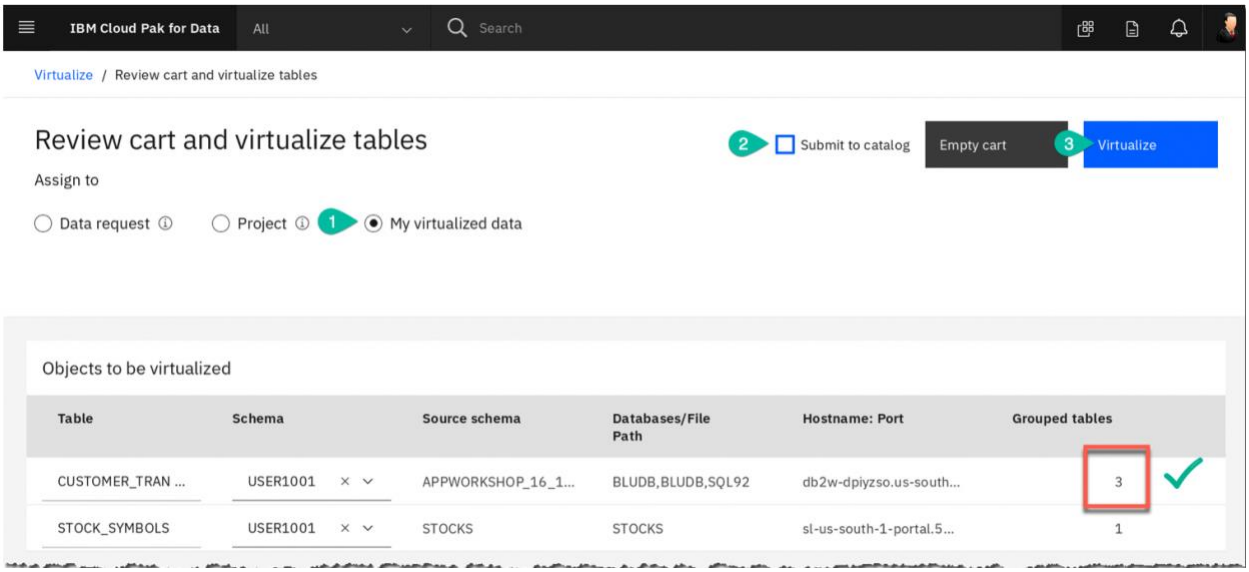
__25. Click View cart.

__26. In the section Review cart and Virtualize tables, notice the Grouped tables column which shows the number 3 to indicate the three CUSTOMER_TRANSACTIONS tables residing in three different databases to be schema folded as a single virtual table.

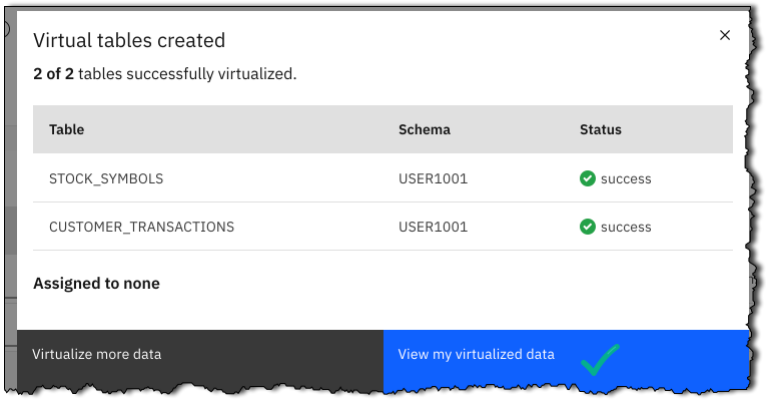
Select the My virtualized data button.

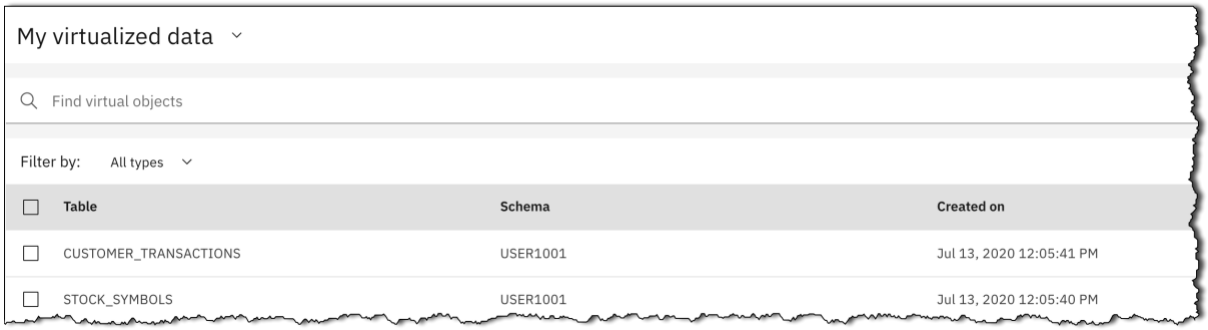
Uncheck the Submit to catalog box.

Click Virtualize.



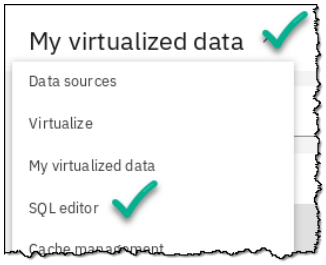
__27. Click View my virtualized data to view the tables created.





__28. You will now be creating a virtualized view of these virtualized tables using the SQL editor.

Select menu (My virtualized data) ⇨ SQL editor.

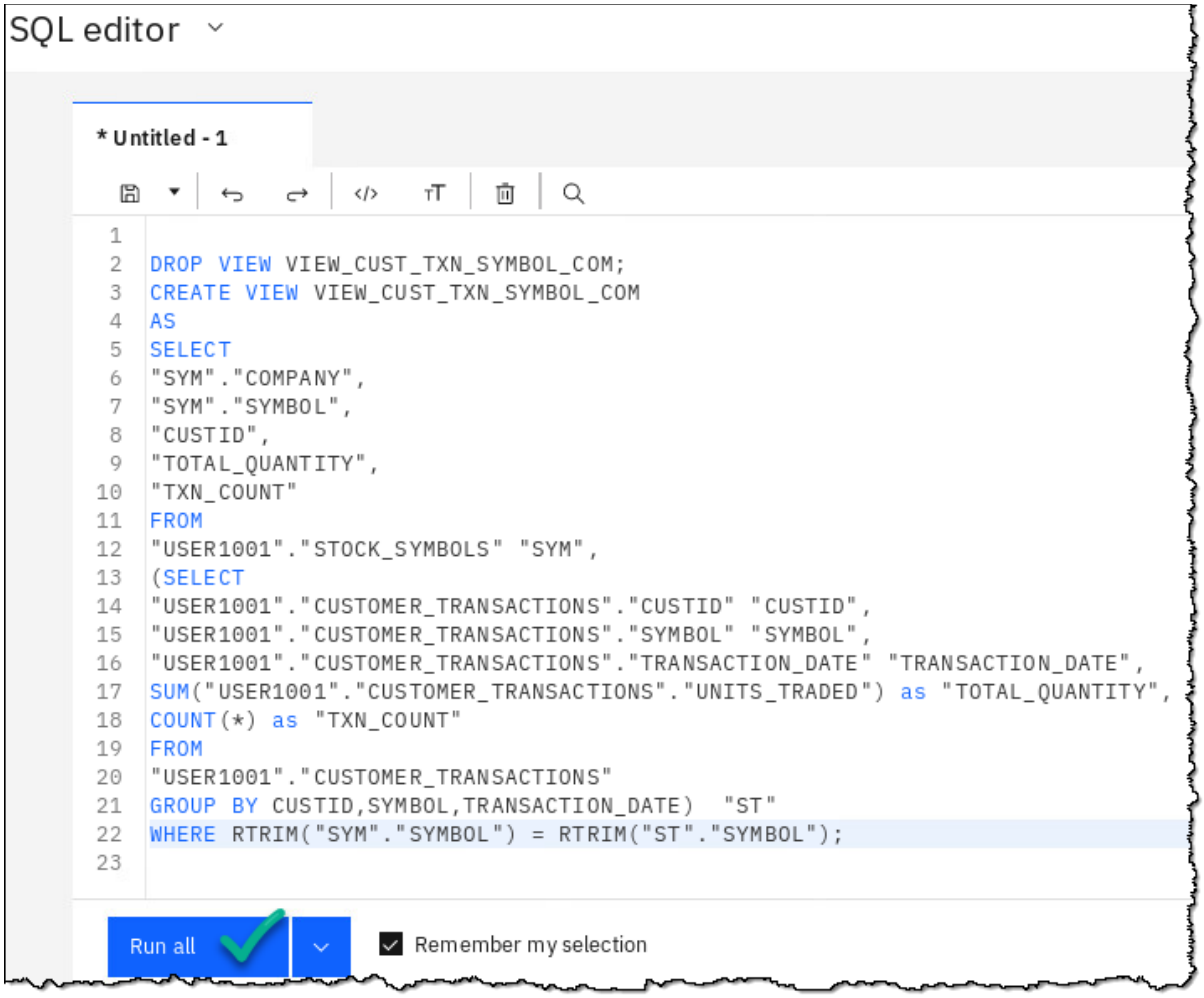


__29. Copy and paste the SQL below into the SQL editor.

 Data Engineer	Note: You can copy and paste this SQL directly from the Unified Desktop by using the File Browser to open this file:  ⇨ Home ⇨ Downloads ⇨ DVCaching_SQL.txt . Alternately, you can download this file by opening a browser tab and using this link: http://ibm.biz/DV-Caching-SQL.

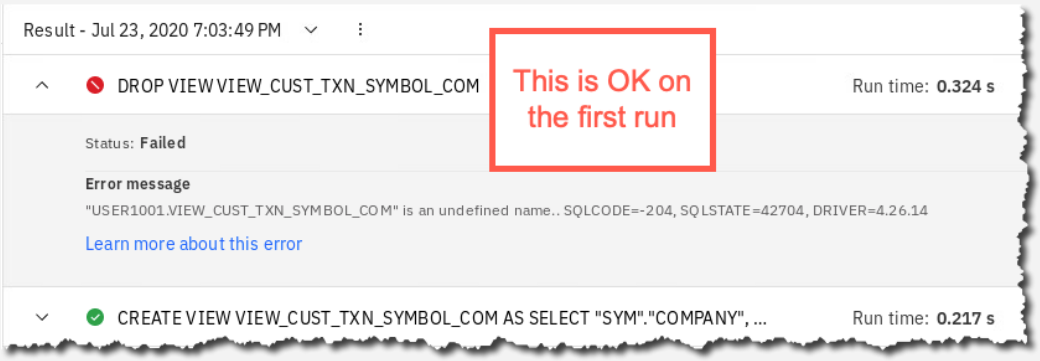
```
DROP VIEW VIEW_CUST_TXN_SYMBOL_COM;
CREATE VIEW VIEW_CUST_TXN_SYMBOL_COM
AS
SELECT
"SYM"."COMPANY",
"SYM"."SYMBOL",
"CUSTID",
"TOTAL_QUANTITY",
"TXN_COUNT"
FROM
"USER1001"."STOCK_SYMBOLS" "SYM",
(SELECT
"USER1001"."CUSTOMER_TRANSACTIONS"."CUSTID" "CUSTID",
"USER1001"."CUSTOMER_TRANSACTIONS"."SYMBOL" "SYMBOL",
"USER1001"."CUSTOMER_TRANSACTIONS"."TRANSACTION_DATE"
"TRANSACTION_DATE",
SUM("USER1001"."CUSTOMER_TRANSACTIONS"."UNITS_TRADED") as
"TOTAL_QUANTITY",
COUNT(*) as "TXN_COUNT"
FROM
"USER1001"."CUSTOMER_TRANSACTIONS"
GROUP BY CUSTID,SYMBOL,TRANSACTION_DATE) "ST"
WHERE RTRIM("SYM"."SYMBOL")= RTRIM("ST"."SYMBOL");
```

__30. Click Run all which should create the virtualized view successfully.

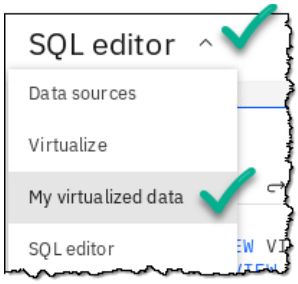


Data Engineer

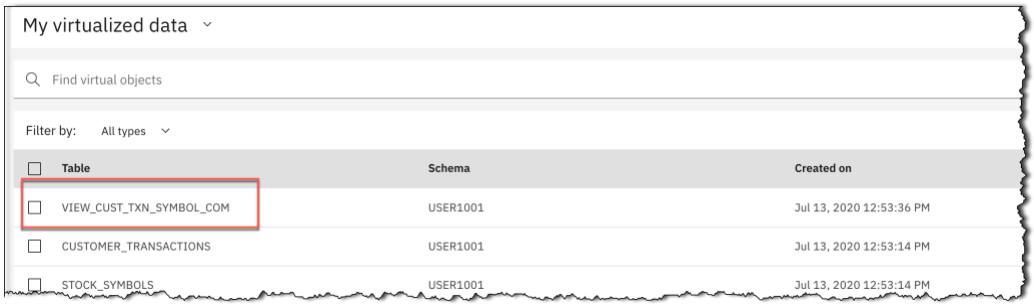
Note: The first time this script is run, the DROP VIEW statement will fail since the view does not exist. It will execute cleanly on subsequent runs.



__31. Select menu (SQL editor) ⇨ My virtualized data.



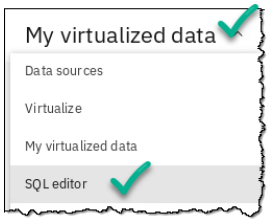
__32. The virtualized view should now be here: VIEW_CUST_TXN_SYMBOL_COM.



12.8.3 Check performance

__33. With the VIEW in place, it would be worthwhile to run a simple aggregated SQL query in the SQL editor to see its performance.

Select menu (My virtualized data) ⇨ SQL editor.



__34. Replace the existing SQL with the following by just typing it in, copy from /Downloads/DV_CheckPerformance.txt file or download from http://ibm.biz/DV-Check-Performance.

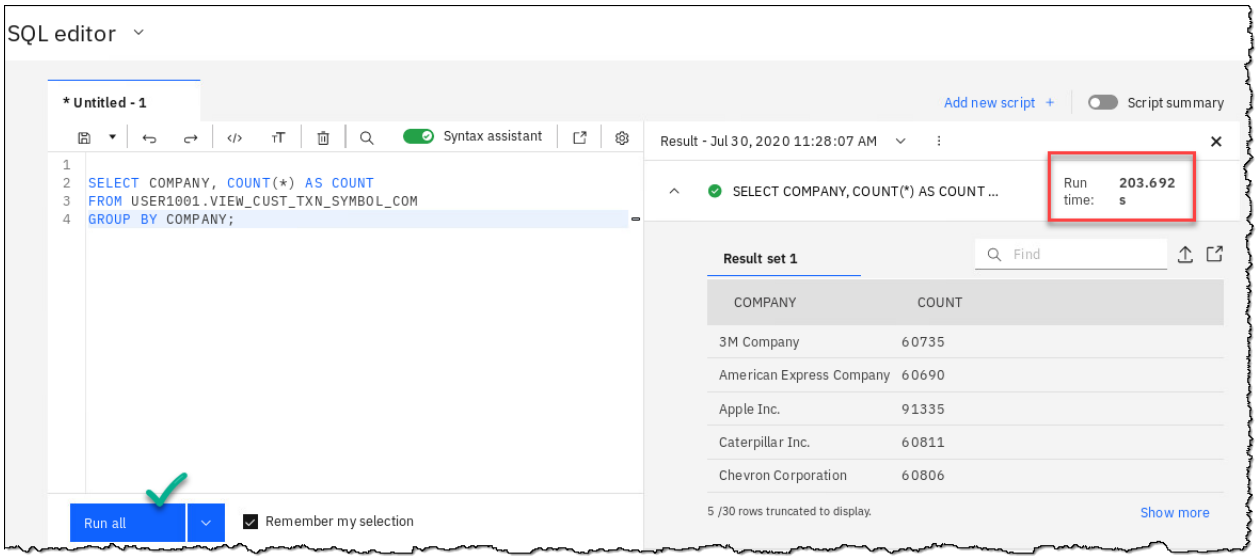
```
SELECT COMPANY, COUNT(*) as COUNT
FROM USER1001.VIEW_CUST_TXN_SYMBOL_COM
GROUP BY COMPANY;
```

(This query performs a simple COUNT across the tables based on the companies traded.)

__35. Click Run all.

The aggregated query is run over the View which in turns gets executed across all three underlying tables in the different data sources via the schema folded virtual table created earlier.

Because we do not have a cache and it is going against all the data in every table, this will take a few minutes to complete – please be patient.



__36. Notice the time taken for running the query.

The time taken for the simple query execution is unacceptable. The Data Engineer reaches out to the DV Admin to create a cache

 Data Engineer	Based on the time taken to execute the simple aggregated query over the View, makes it a perfect candidate to create a DV data cache from and that will be the next step. If this step fails, check to make sure that you have copied the SQL code properly in the SQL editor and run it again. Keep in mind that a successful run of the SQL may not mean it was actually created properly.

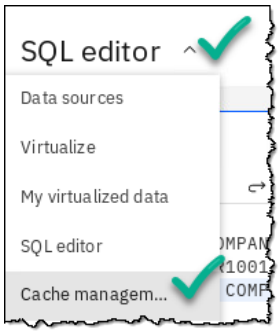
12.9 Creating the DV data cache

Given the performance issues observed during the SQL execution, creating a DV data cache from the View is a prudent approach.

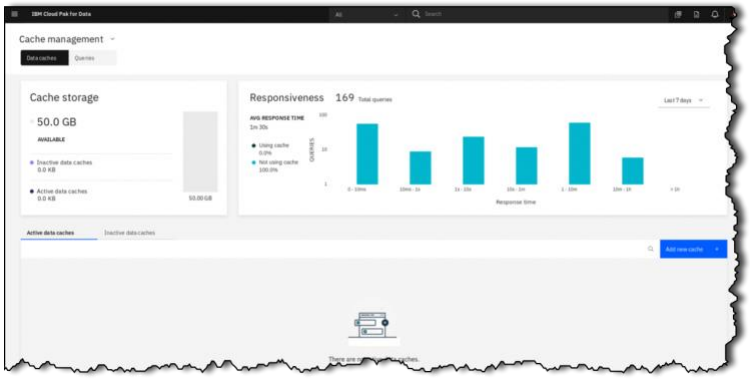
Persona (Role)
 Administrator (DV/CPD)

The DV Admin can analyze the queries executed against DV and identify those generated by the dashboard. Looking at the query will help understand what type of cache needs to be created to improve performance for a better dashboard experience.

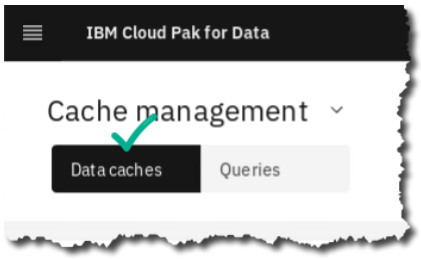
__37. Click Menu (SQL editor) ⇨ Cache Management.



__38. You should be presented with your Cache Management Dashboard.



__39. Click on Data caches.

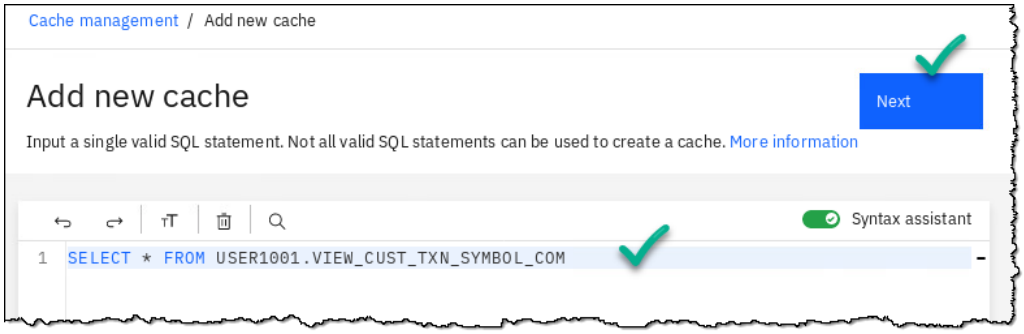


__40. Towards the bottom right of the screen, find and click Add new cache +.

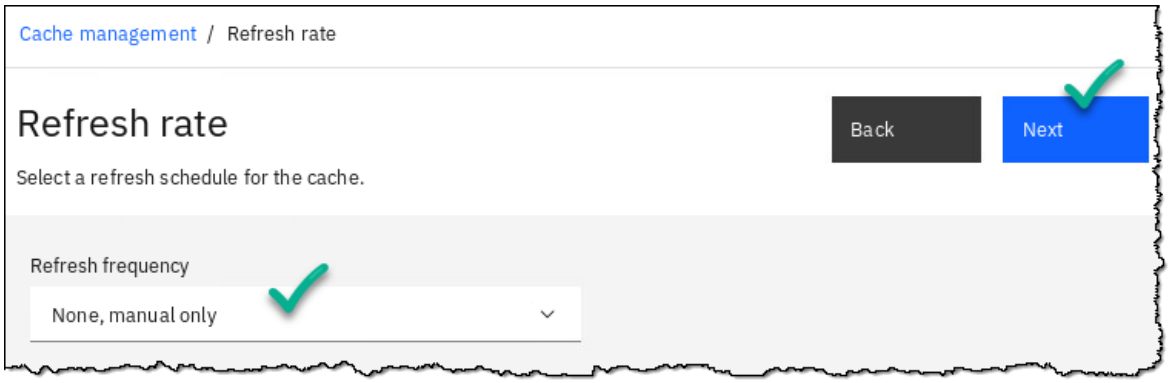
__41. This opens a SQL Editor to type in the query from which the cache will be created. Type in the query to select all columns from the virtualized view:

SELECT * FROM USER1001.VIEW_CUST_TXN_SYMBOL_COM

Click Next.

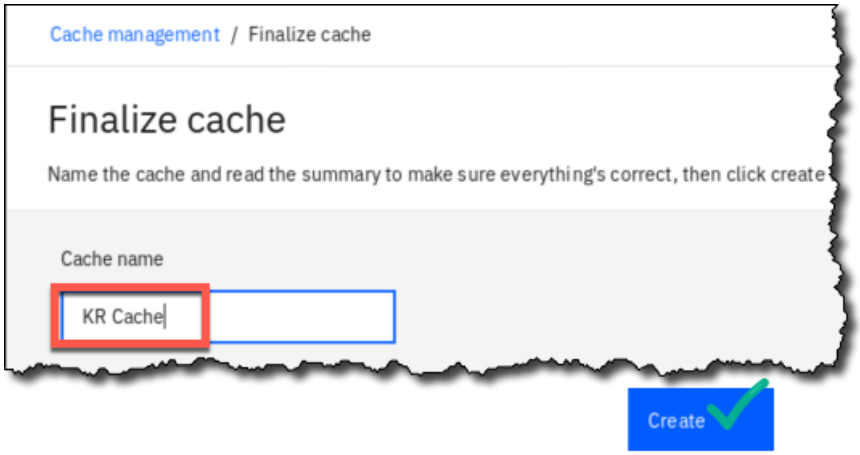


__42. In the *Refresh rate* choose the default (None, manual only) and click Next.

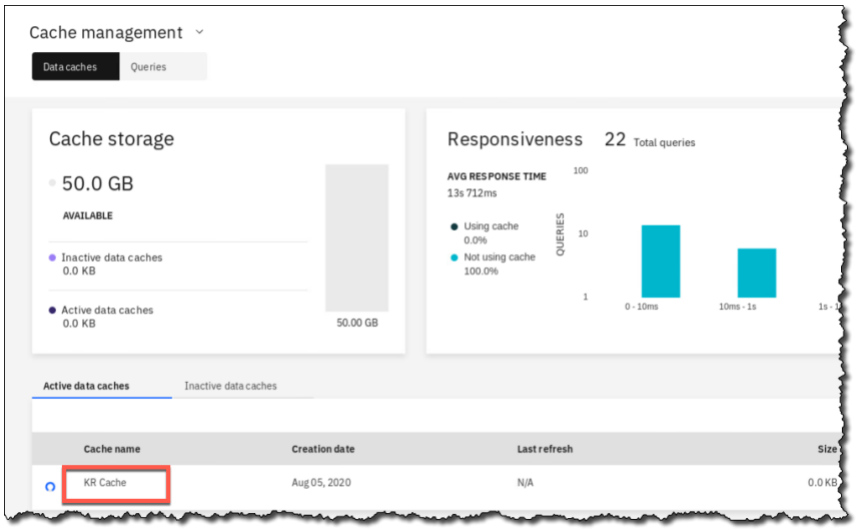


__43. Confirm the details on the final page and click Create.

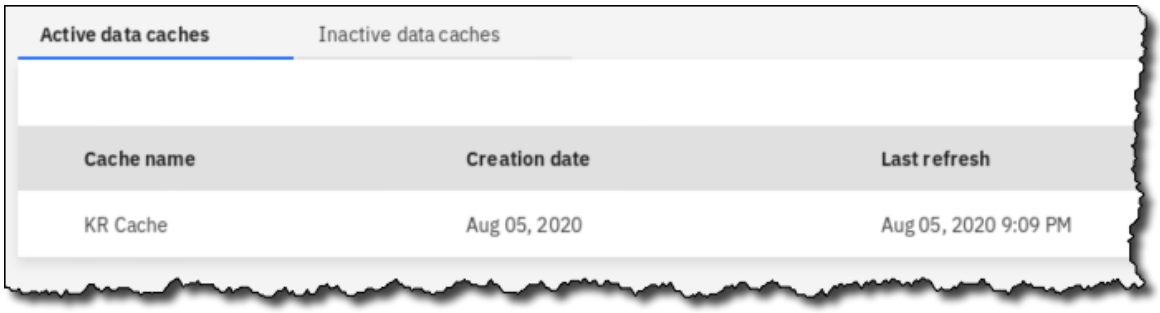
(Note: Your cache name will vary from what is shown below.)



__44. The cache creation process may take some time and the main Cache Management page will reflect the work in progress.



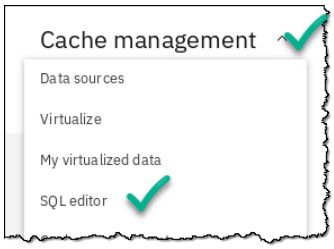
__45. Once the cache creation is complete, the newly created cache shows up under the Active data caches along with other details and its size.



12.9.1 Flush the package cache and updating the statistics

Admin	KNOWN ISSUE – There exists a defect/issue wherein a query executed before the cache is created and activated fails to use the cache. This is because the query plan is already cached in a relational database such as Db2, so the work around is to clear the Db2 package cache and collect table statistics on the virtual tables created above. You will do that next.

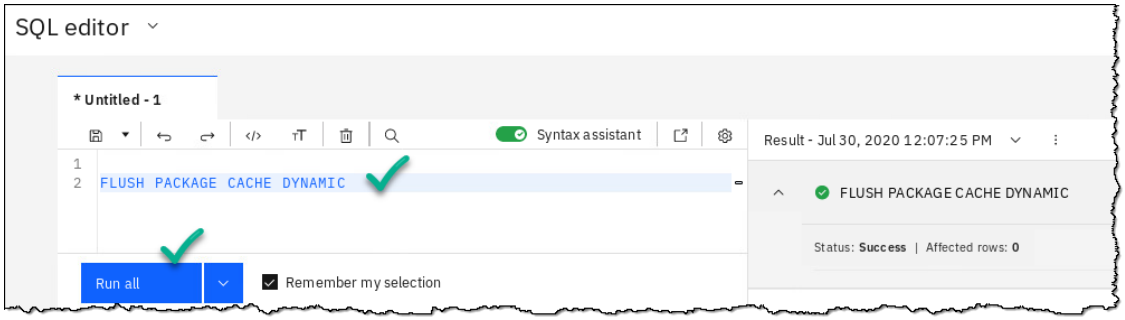
__46. Click Menu (Cache management) ⇨ SQL Editor.



__47. Remove the view SQL and replace it with the following:

FLUSH PACKAGE CACHE DYNAMIC

__48. Click Run all.



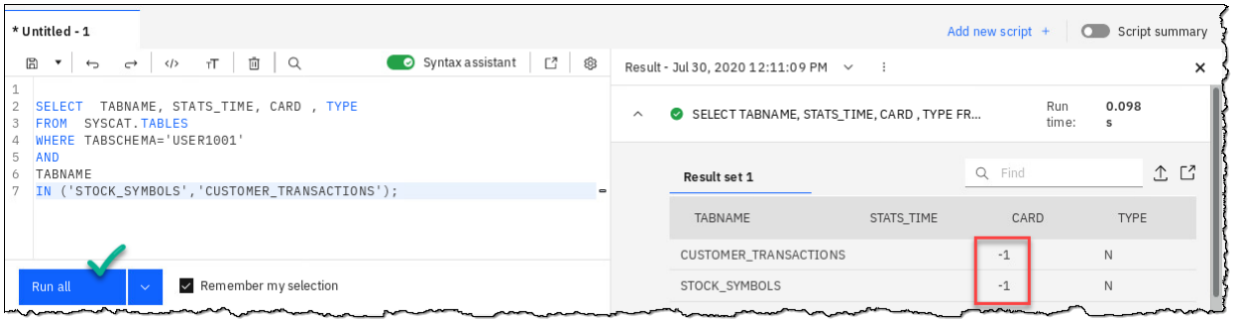
In a production setting, flushing the entire package may not always be a good idea as it could affect query execution for all users of the database. (We did it here to make this lab simpler.)

Instead, one could opt to selectively flush the package corresponding to particular queries. For more details on that see: http://ibm.biz/FLUSH-PACKAGE.

__49. Next, check if the table statistics have been collected. Replace the previous SQL with the following: (You can also download this SQL from http://ibm.biz/DV-Select-Tab.).

```
SELECT  TABNAME, STATS_TIME, CARD , TYPE
FROM  SYSCAT.TABLES
WHERE TABSCHEMA='USER1001'
AND
TABNAME
IN ('STOCK_SYMBOLS','CUSTOMER_TRANSACTIONS');
```

__50. Click Run all.



If the cardinality for the virtual tables created before shows -1, this indicates that table statistics have not been collected. These statistics will be collected now.

Note: the SYSPROC.NNSTATS is a procedure to collect statistics for remote tables: http://ibm.biz/SYSPROC-NNSTATS.

__51. If your card shows -1, then perform this step. If not, then you can skip this step.

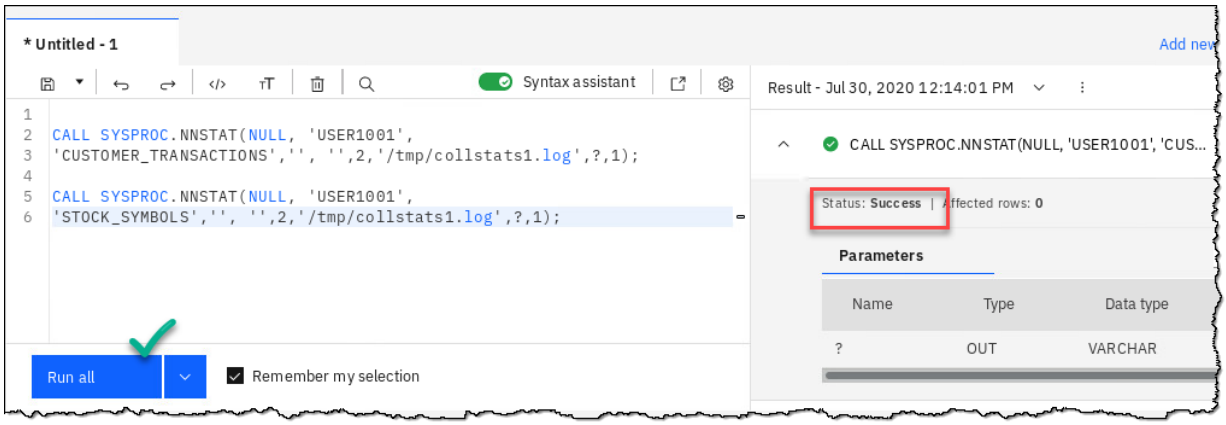
Replace the previous SQL with the following:

(Or use SQL found here: http://ibm.biz/DV-Call-SYS)

CALL SYSPROC.NNSTAT(NULL, 'USER1001',
'CUSTOMER_TRANSACTIONS','', '',2,'/tmp/collstats1.log',?,1);

CALL SYSPROC.NNSTAT(NULL, 'USER1001',
'STOCK_SYMBOLS','', '',2,'/tmp/collstats1.log',?,1);

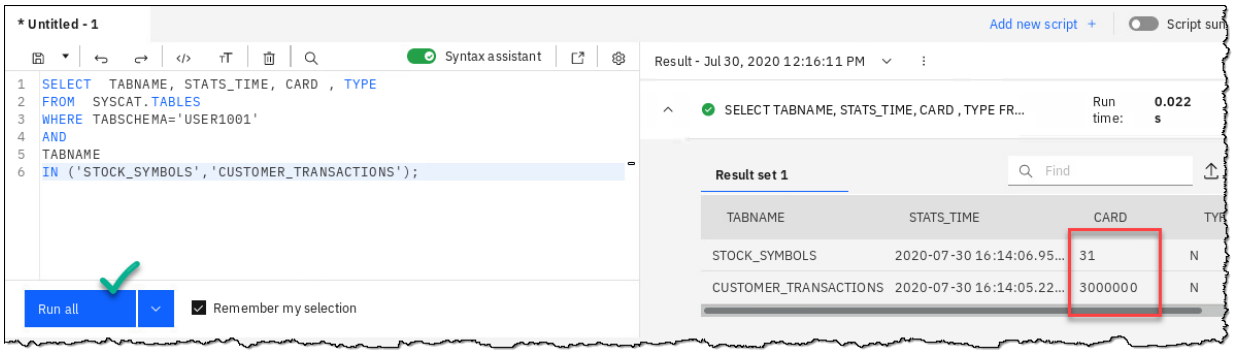
__52. Click Run all.



__53. To confirm if the table statistics have now been collected, replace the previous SQL with the following (use the SQL from before):

SELECT TABNAME, STATS_TIME, CARD , TYPE
FROM SYSCAT.TABLES
WHERE TABSCHEMA='USER1001'
AND
TABNAME
IN ('STOCK_SYMBOLS','CUSTOMER_TRANSACTIONS');

__54. Click Run all.



The actual row counts for the virtual tables are shown as expected instead of -1.

__55. With the package cache flushed and the table level statistics collected, all queries henceforth referencing the virtualize view should start using the cached copy instead of accessing the underlying data sources directly.

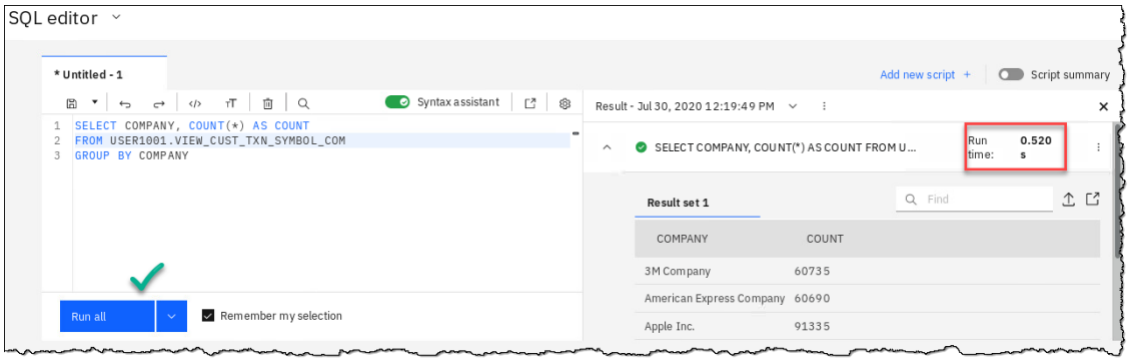
Re-run the previously execute aggregated query on the View:

Remove the previous SQL and type in this again:

SELECT COMPANY, COUNT(*) as COUNT
FROM USER1001.VIEW_CUST_TXN_SYMBOL_COM
GROUP BY COMPANY;

This query performs a simple COUNT across the tables based on the companies traded.

__56. Click Run all.



The Data Engineer the informs the BA of the cache creation and shares the View and other relevant details with the BA to continue the dashboard work.

12.10 Lab conclusion

Data Virtualization (DV), as part of the Collect phase, facilitates accessing data from various data sources such as DVM for z/OS and performs queries across them. Since data movement is limited, all the access rules and the policies for creating copies remain preserved. In situations where query response times are paramount, DV provides the caching facility, which was covered in this lab.

**** End of Lab 12 –** Collect: Virtualizing & Caching from z/OS – Deeper Dive

Lab by Dave Trotter and Rajesh Kartha, Edited by Burt Vialpando and Kent Rubin - IBM

Lab 13 ORGANIZE – DEEPER DIVE

13.1 Lab overview

Many organizations find it difficult to understand their own data because it originates from many sources, is dispersed across many silos, and is controlled by different teams.

This Organize lab will show you how to uncover the hidden data in your organization's data and how to build a lineage that is otherwise difficult to establish. Cloud Pak for Data helps your organization move from the manual processes required to establish relationships between data to an automated one aided by the platform's built-in machine learning capabilities.

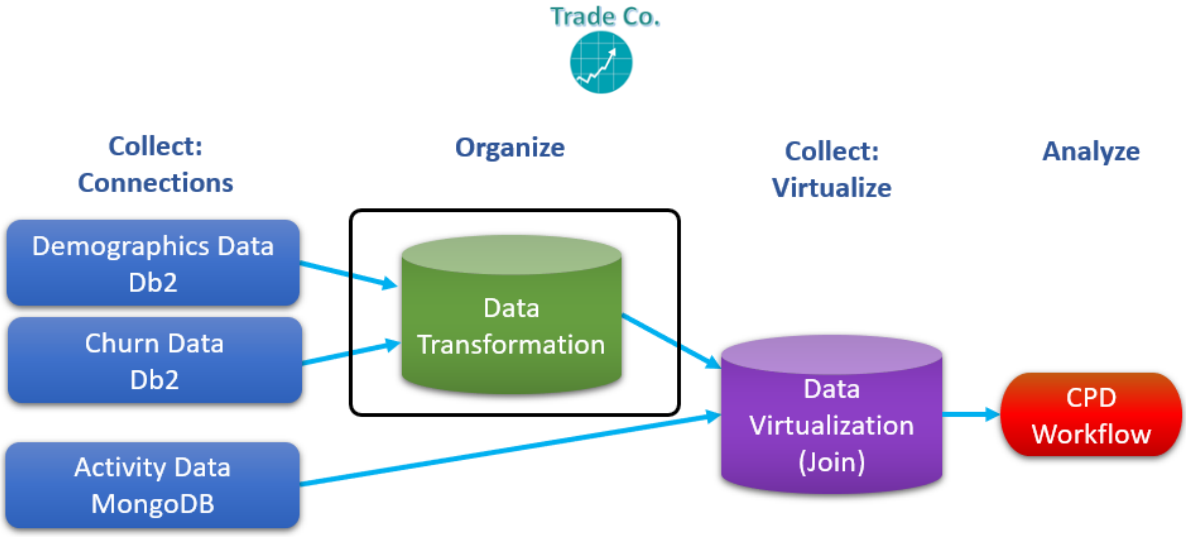
13.2 Personas represented in this lab

The Data Steward and Data Engineer personas are most likely to perform most of the Organize tasks shown in this lab.

Persona (Role)	Capabilities
 Data Steward	Data Stewards integrate and transform the data, and provide governance, lineage and classification of the data.
 Data Engineer	Data Engineers build and optimize the systems to allow data scientists and business analysts to perform their work. The Data Engineer ensures that any data is properly received, transformed, stored, and made accessible to other users.

The Data Steward and Data Engineer personas often work closely together to prepare the data for analytics processing by other personas. For example, in this lab one of the things the Data Steward will do is to ensure Business Terms are assigned to data assets. In turn, the Data Engineer will then use that information to find the appropriate data to extract and transform to create a final table of the data sources joined together.

The Data Steward persona also works closely with the Data Quality Analyst persona.



13.3 Logging into the CPD web client (if you have not already done so)

__1. If you are starting this lab stand-alone (without going through previous labs) do the following:

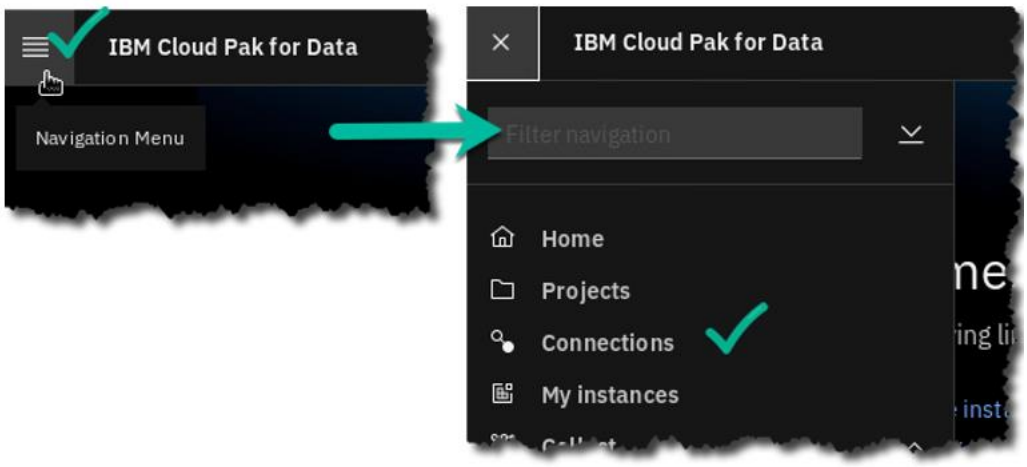
__2. Double-click the desktop icon: Cloud Pak for Data Web Client.

__3. The CPD web client GUI displays as shown. Use cpduser and cpdaccess for the *Username* and *Password* and click Sign In.

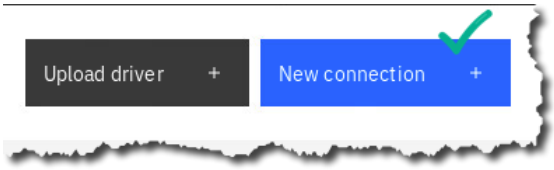
13.4 Creating a connection to your data

Data Connections are one of the primary ways that Cloud Pak for Data can access data both within the application and across various sources, including cloud, on-premises, application, semi-structured, etc. These connections can be created globally and then used by users within the different capabilities of the CPD platform.

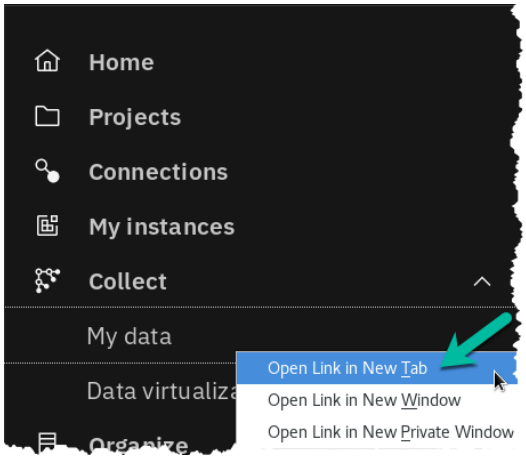
__4. If you are starting this lab stand-alone (without going through previous labs) do the following: In the CPD web client, start at the Navigation Menu ⇨ Click Connections.



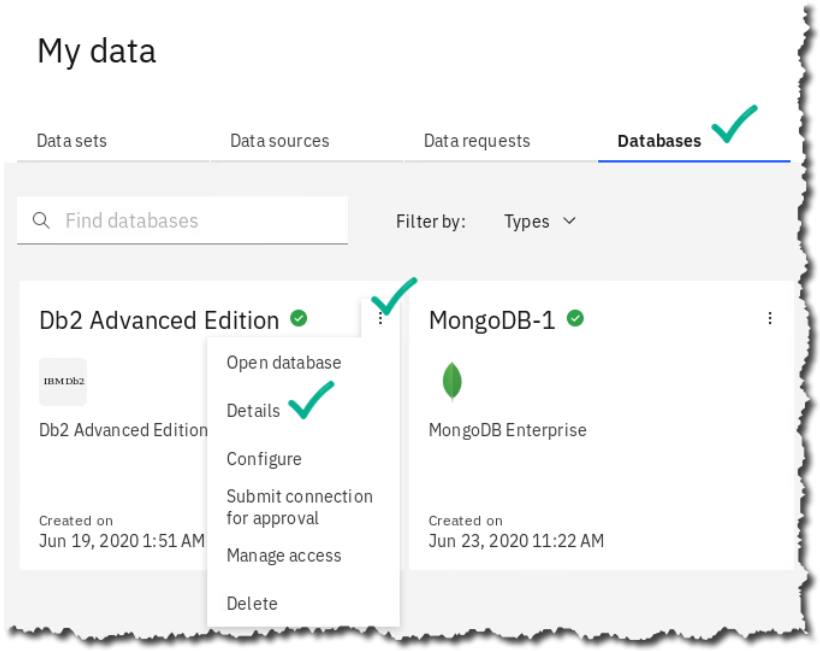
__5. Click New connection.



__6. Before creating a connection, you will need to know the credentials to the database. You can find this by clicking on Navigation Home, Collect, then right click on My data and Open Link in New Tab.

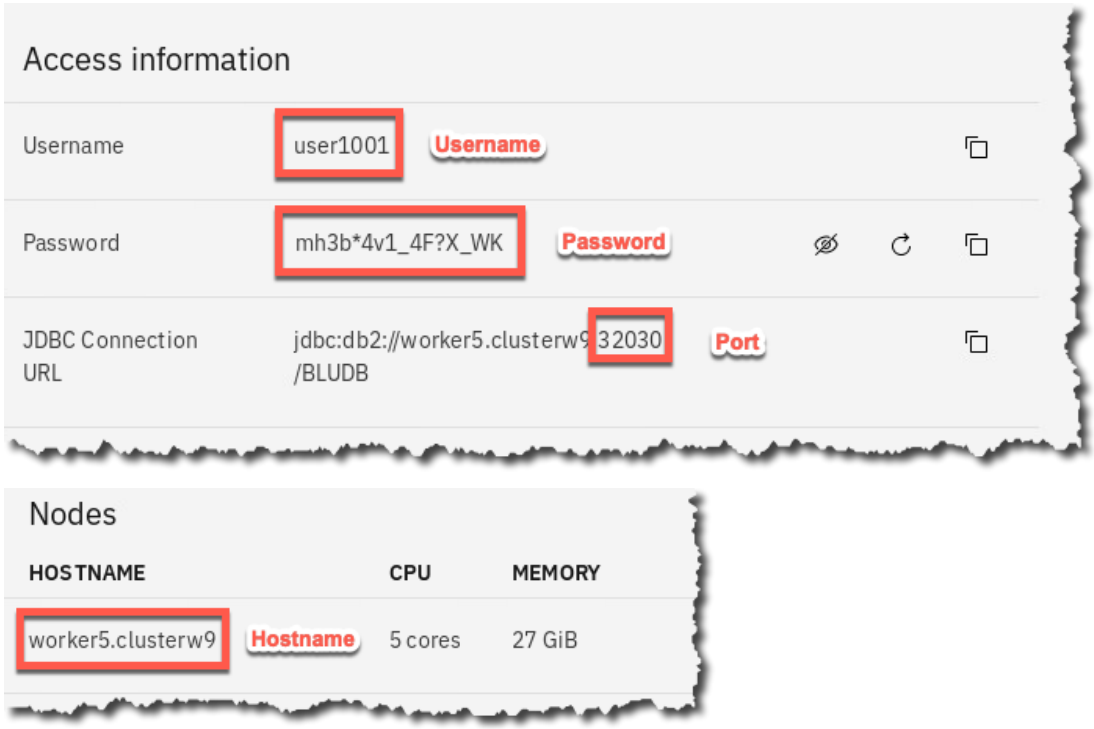


__7. Click on the tab Databases ⇨ Click on the tile Db2 Advanced Edition Ellipse ⇨ Details.

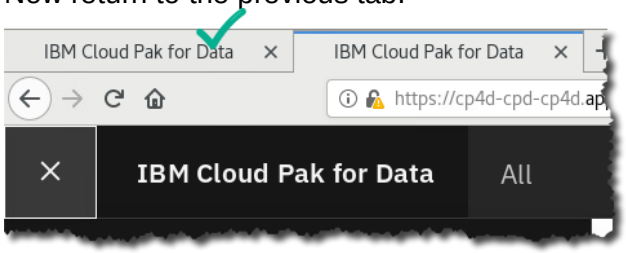


 Data Steward	One of the advantages of using a full platform that uses containers and microservices is that you can have a host of capabilities that can be turned on, and with the correct user privilege, instances of those capabilities (like a new MongoDB or Db2 instance) can be created with the click of a button – just as you would be able to do in a Public Cloud, but behind your firewall.

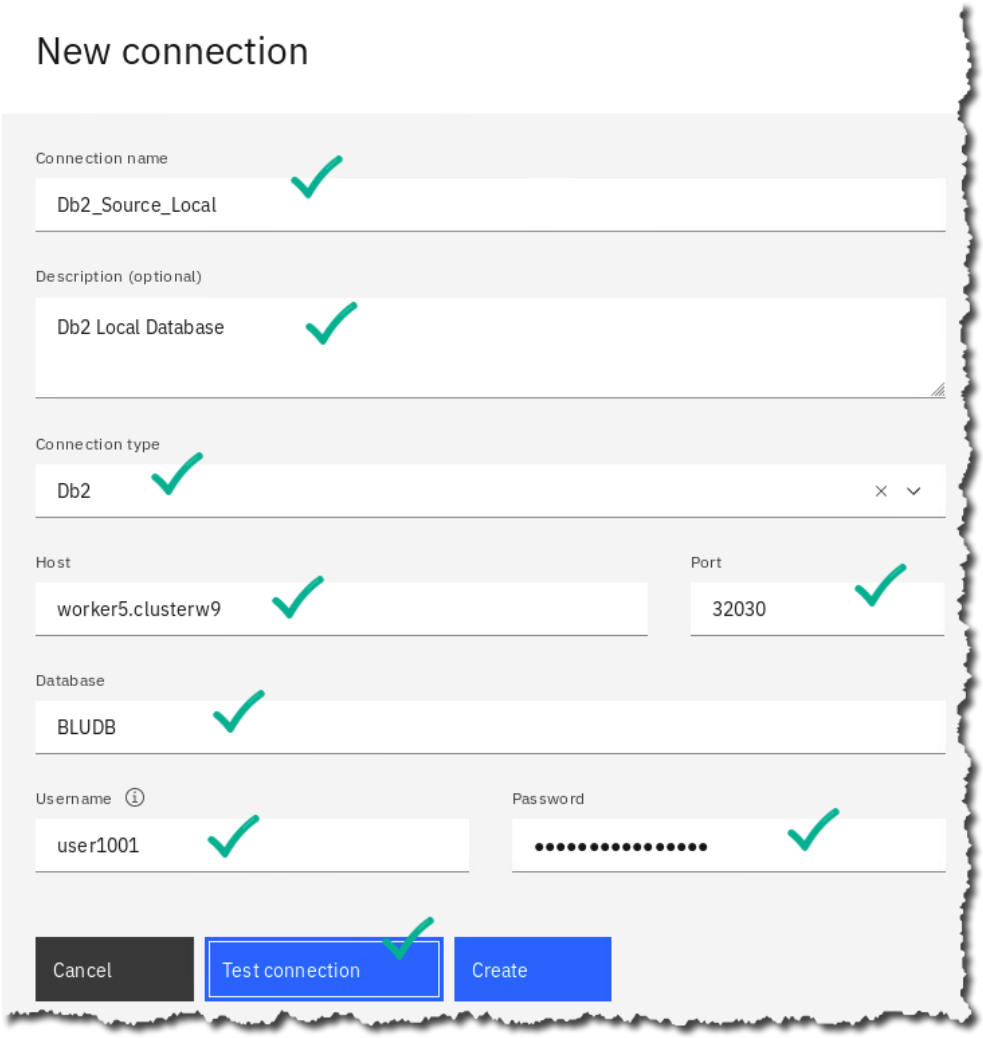
__8. Page down to the bottom of the screen and you'll see the access information for our database. Copy down this information to enter in our Data Connections window.



__9. Now return to the previous tab.



__10. Enter the parameters to create a connection to the internal Db2 database used for this workshop. Below is a sample; Connection Name and Description can be whatever you choose, the rest of the credentials you can get from the previous step.

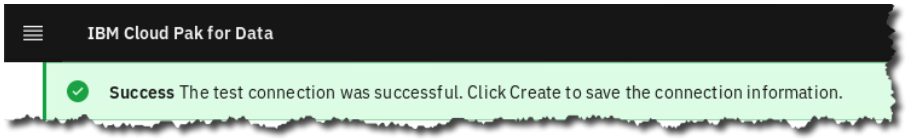


Data Steward

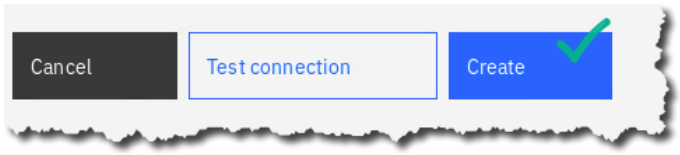
Note: You may copy and paste the password as shown below. (if this doesn't work, then try highlighting the password and use [CTL]-c then [CTL]-v).

__11. Click Test connection (to verify your credentials are correct).

__12. That will return a successful message if the connection parameters are correct.



__13. Click the Create button to add the connection to CPD. At this point it is a ‘Global Connection’. We will later add it to different projects, catalogs, etc.



13.5 Working in a project

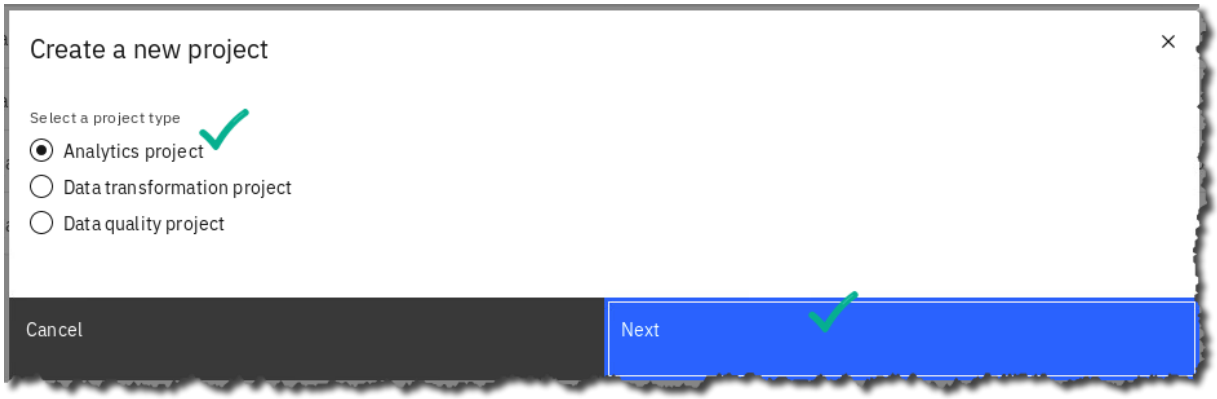
Working with Projects is an important part of the Cloud Pak for Data experience. Projects allow you to organize your work into specific areas to which you can control access, and once you are happy with the results, you can publish the assets and the findings to the Catalog for general user consumption.

__14. Let's start by creating a project:

In the CPD web client, click the Navigation Menu ⇨ Projects ⇨ Click New project.

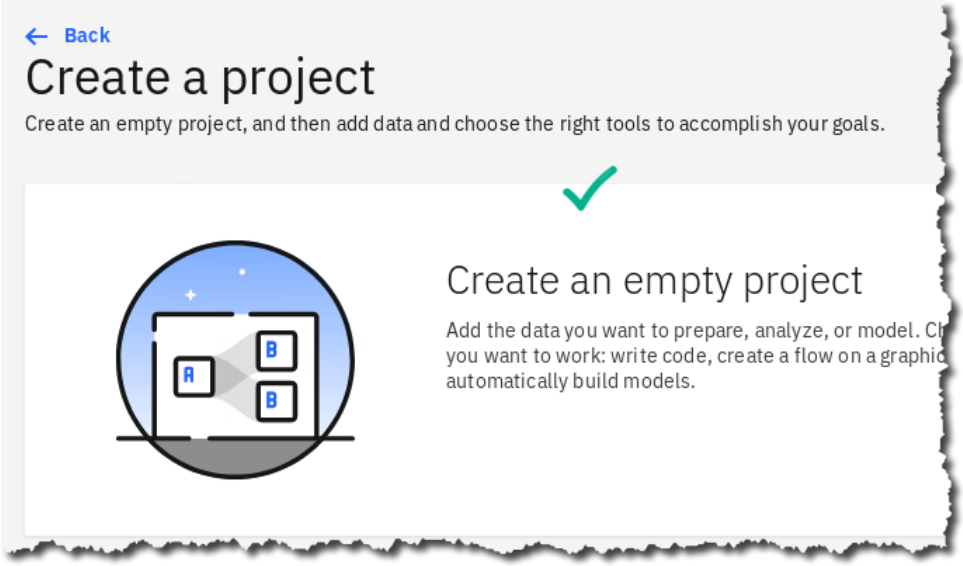
__15. There are three primary types of projects you can create.

Click Analytics project ⇨ Next.

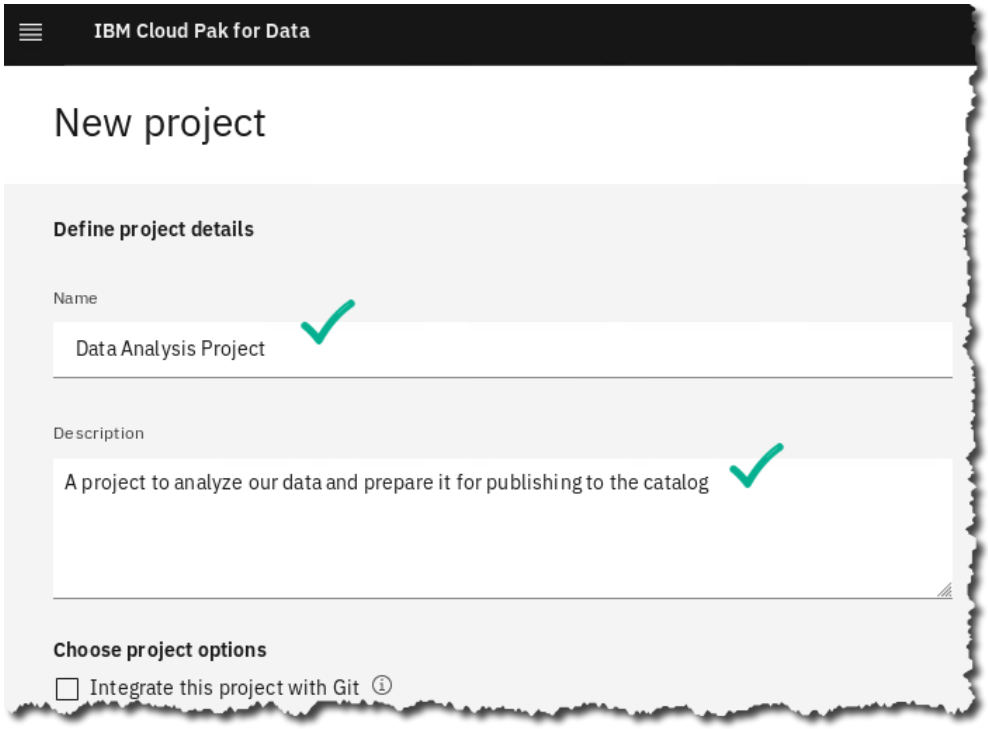


__16. You can create an empty project,create a project by importing existing assets from your file system, or from a Git repository.

Click tile Create an empty project.



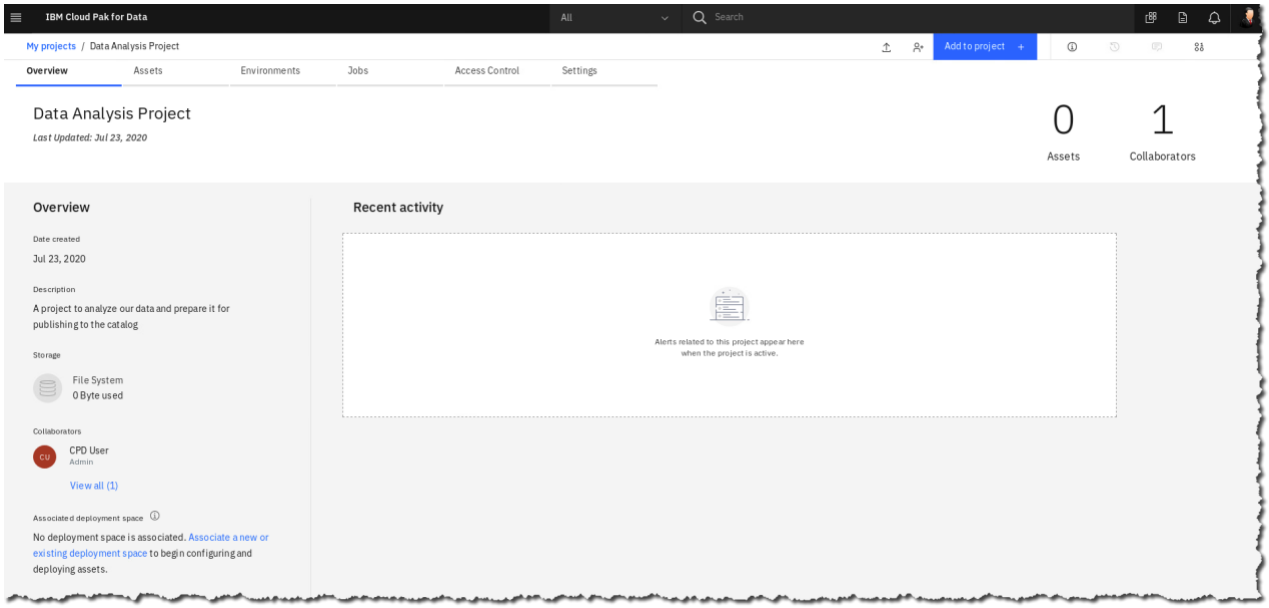
__17. Add the Name and Description for your project.



Data Steward	Note: You can also integrate the project to Git, which allows for automatic saves to a Git repository – we won't be doing that today.

__18. Click Create.

This will create a project and put you into the project overview screen.

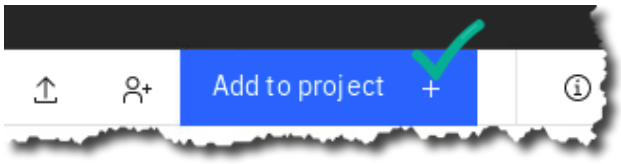


This screen shows summary information about who is a collaborator on this project, what assets have been created or exist in this project, etc. Along the top you see tabs for the different types of things we'll do with this Project, for example, Assets, Environments, etc.

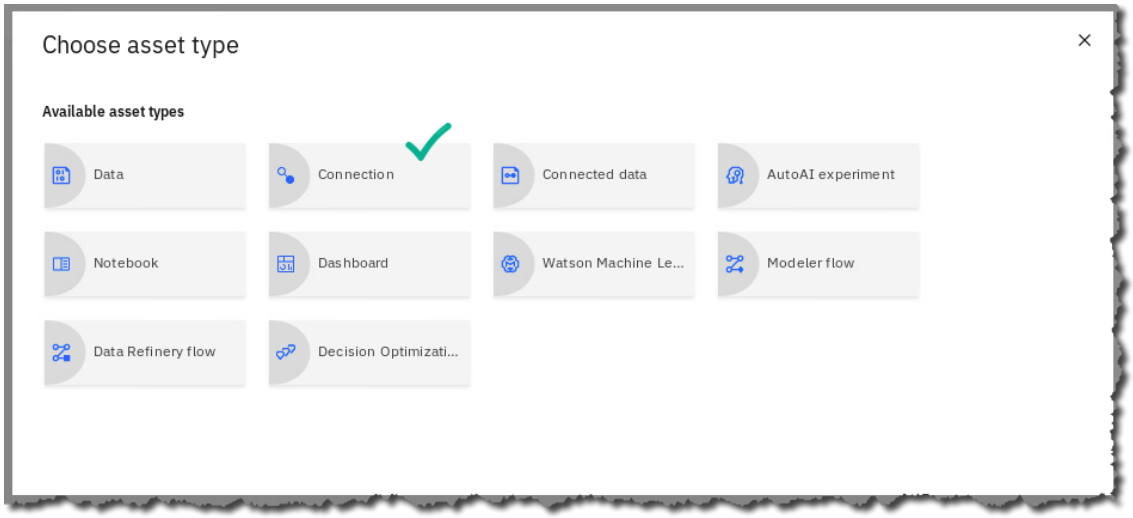
13.5.1 Adding a connection

Next, we'll start to bring data into our project so we can start to analyze, understand and enhance our data, to get it ready for publishing to a Catalog for public use.

Click Add to project.

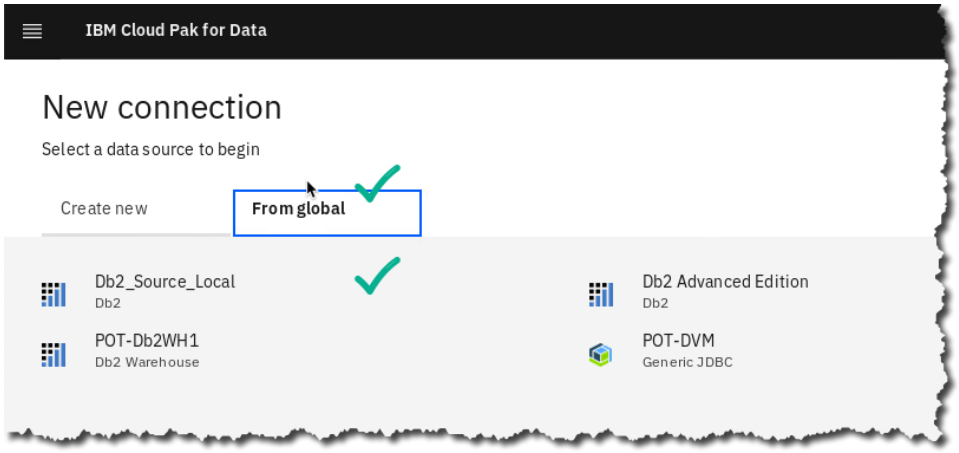


__19. Choose Connection, and we will add the connection we created earlier.



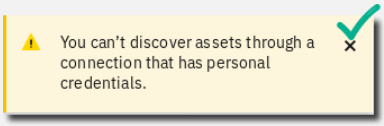
__20. Here, you see a number of different choices for creating a connection. This is one of the strengths of CPD – the breadth of different types of data you can access directly from the platform. As we said, we're going to add the connection we created earlier.

Choose From global ⇨ Db2_Source_Local.



Data Steward

Note: If this message appears, click the X



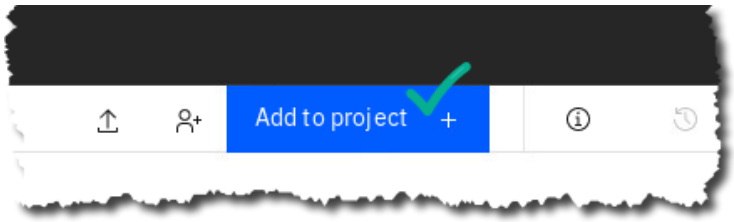
__21. Click Test (to validate we can connect to our data source),

then click Create.

13.5.2 Adding connected data

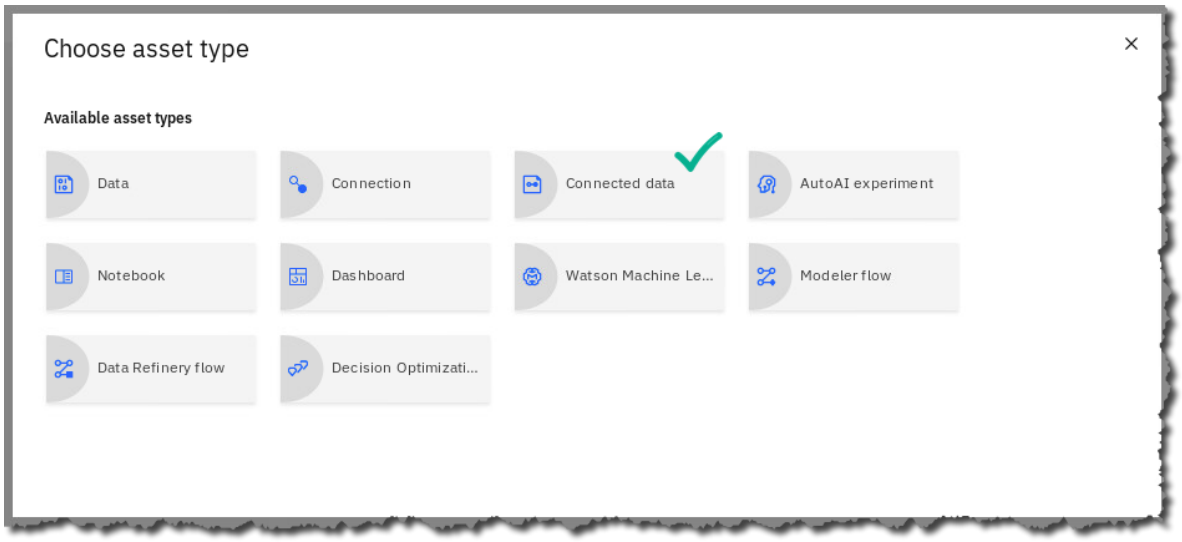
__22. You've now added your first asset to the project (a connection). Next we'll want to use that connection to bring in some data and start to analyze and understand it.

Once again, click Add to Project**.**

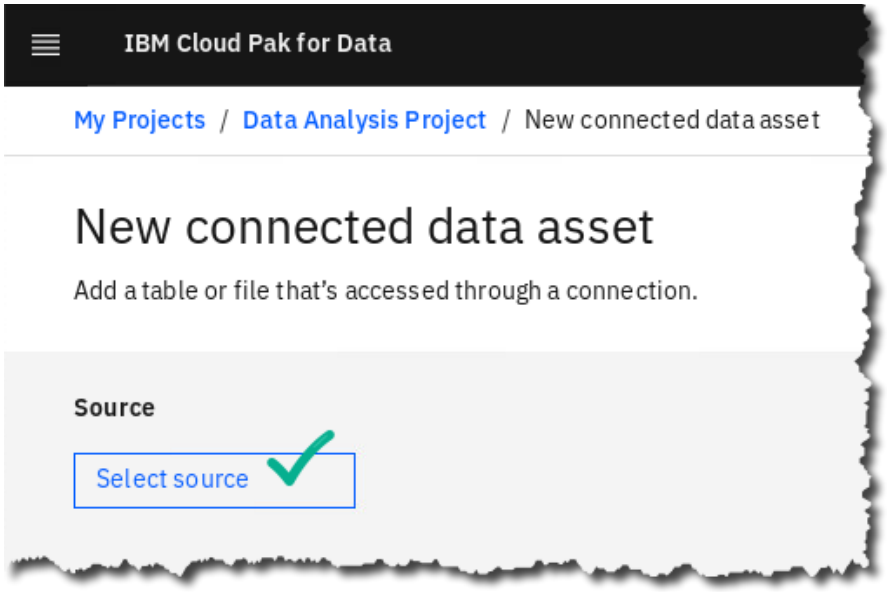


__23. Notice you can also add files directly from your desktop, by either dragging or browsing your file system.

Now choose Connected data.



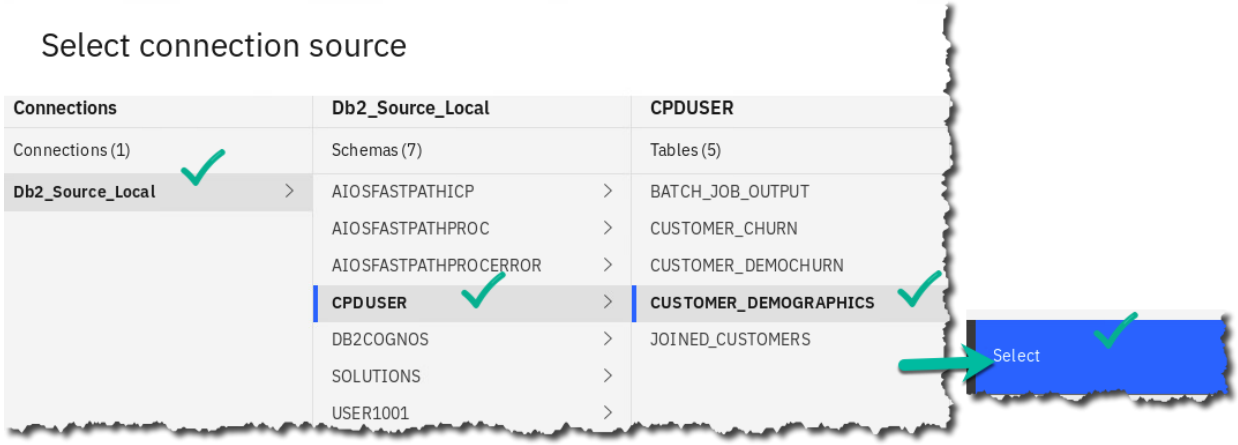
__24. Click Select source.



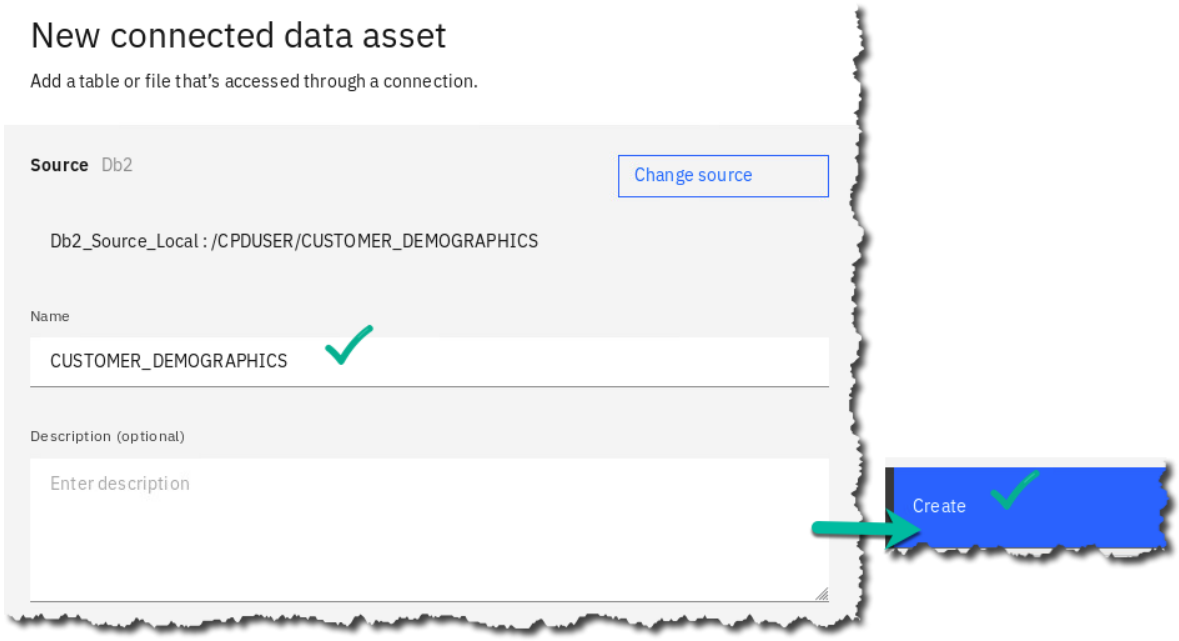
__25. Next, we'll choose our Connection, Schema, and Table(s). First, click Db2_Source_Local (or the name of your connection).

We are interested in two tables, that are in two different schemas.

Choose the schema CPDUSER ⇨ CUSTOMER_DEMOGRAPHICS ⇨ Select.

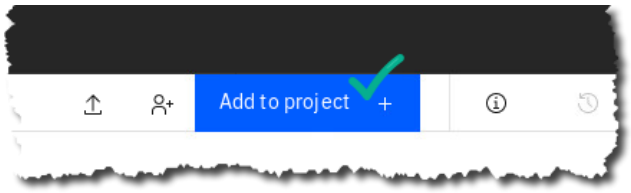


__26. Now we'll give it a name to refer to it by in our project. Enter CUSTOMER_DEMOGRAPHICS for the Source name ⇨ Create.

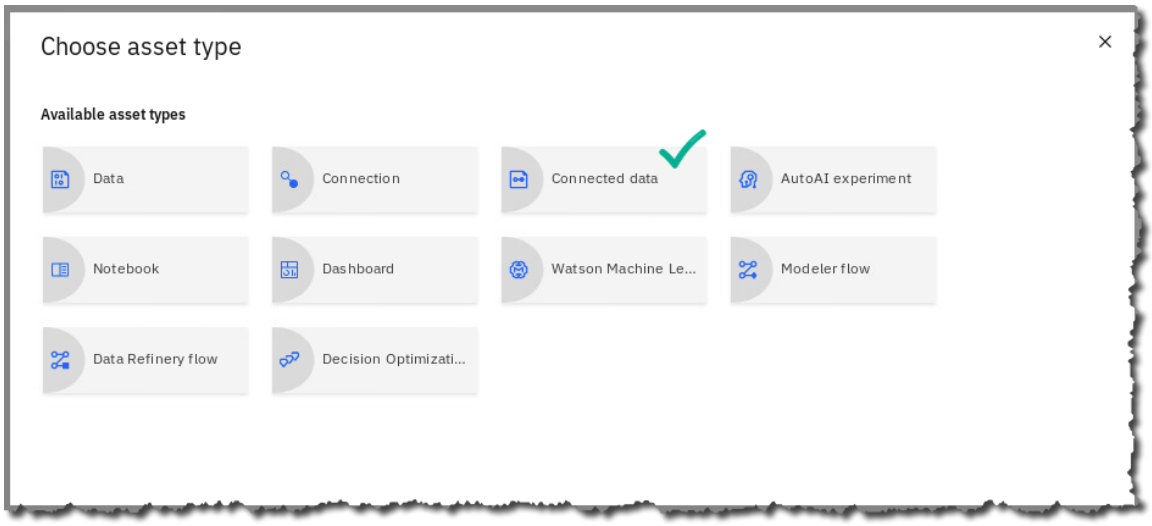


__27. You've now added your first data asset to the project! Let's add a second table from another schema.

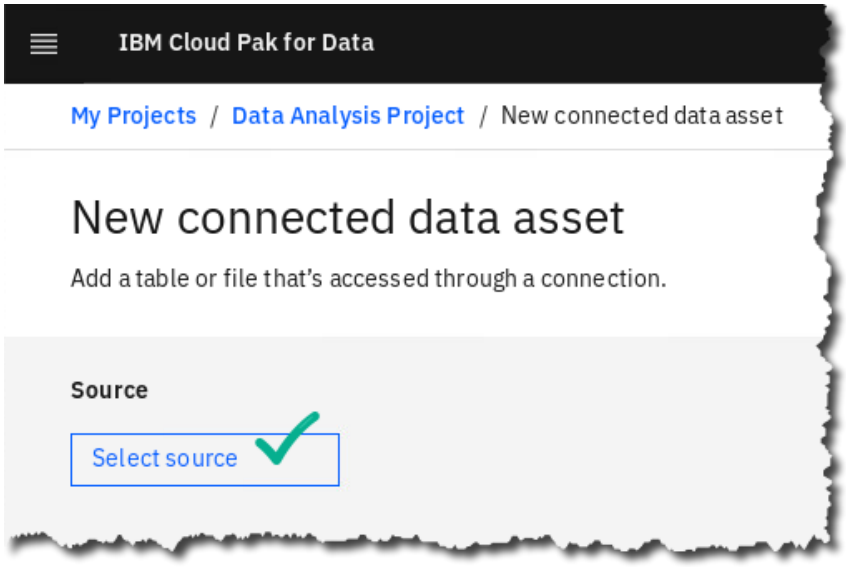
Once again, click Add to Project.



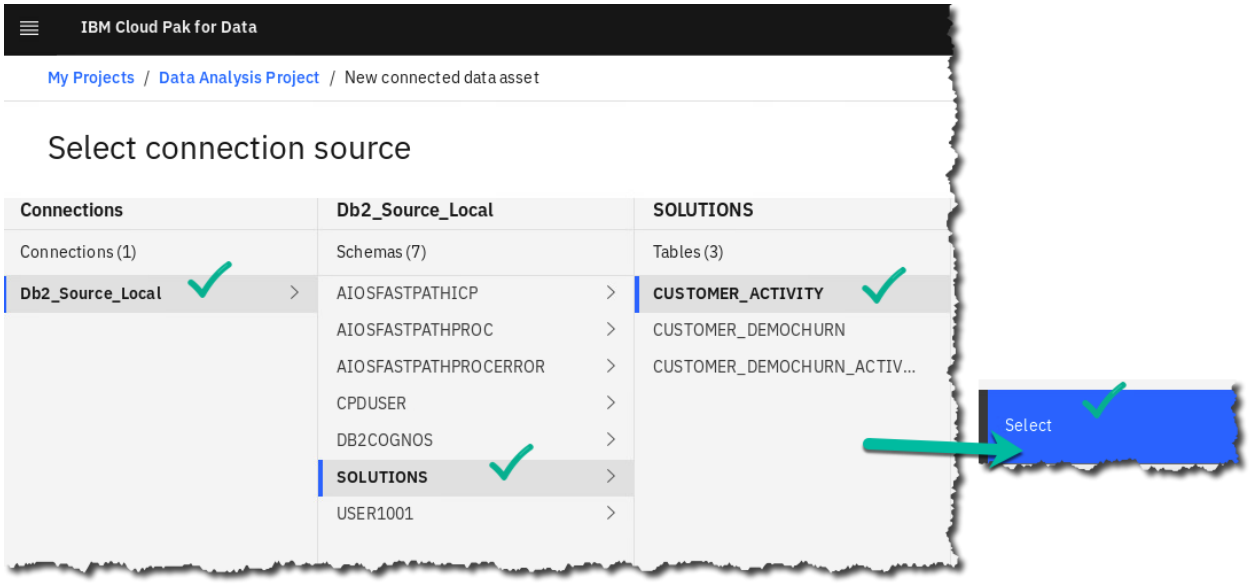
__28. Choose Connected data again.



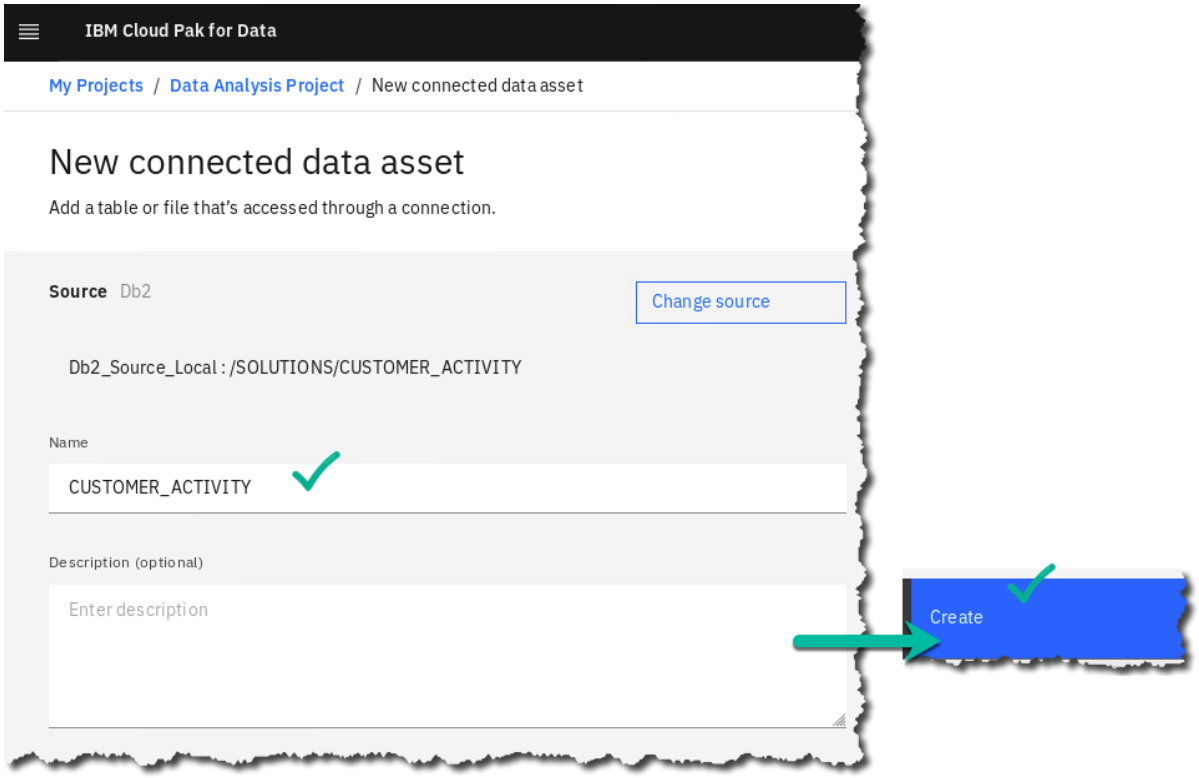
__29. Click Select source.



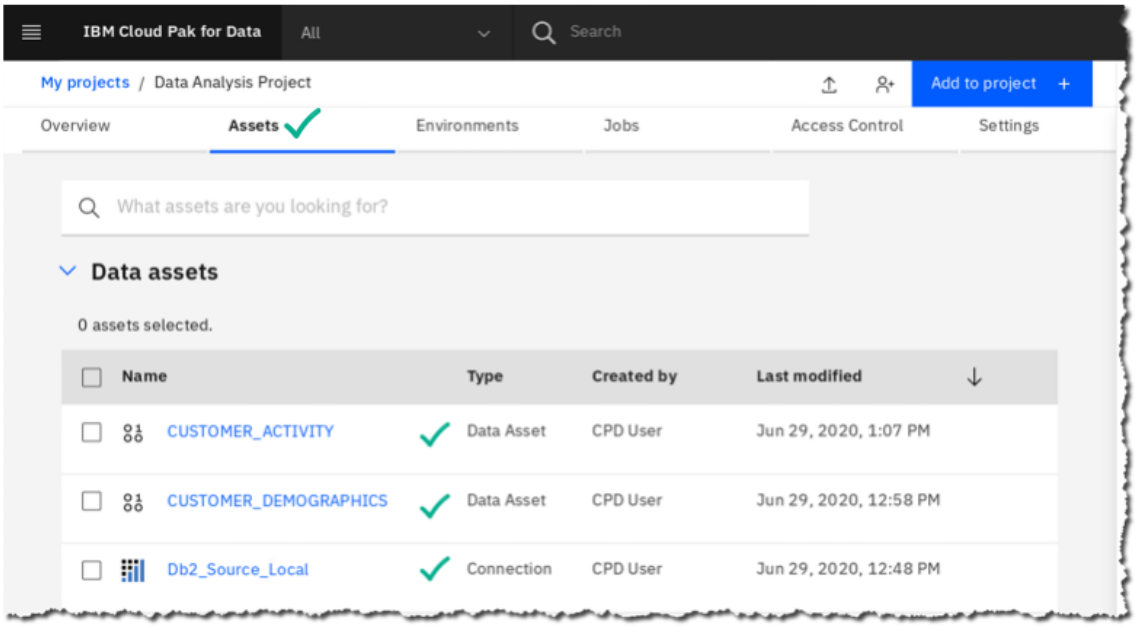
__30. Now select – Db2_Source_Local ⇨ SOLUTIONS ⇨ CUSTOMER_ACTIVITY ⇨ then Select.



__31. Name the asset CUSTOMER_ACTIVITY ⇨ Create.



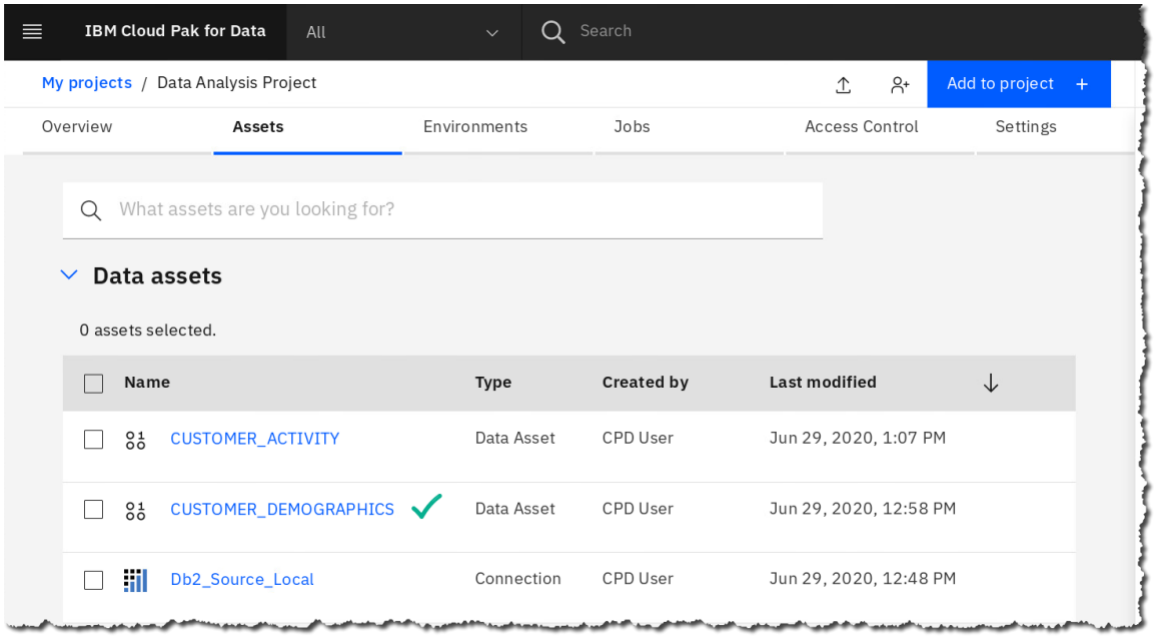
__32. When complete, your assets list for your project should contain 3 assets; two tables and a connection.



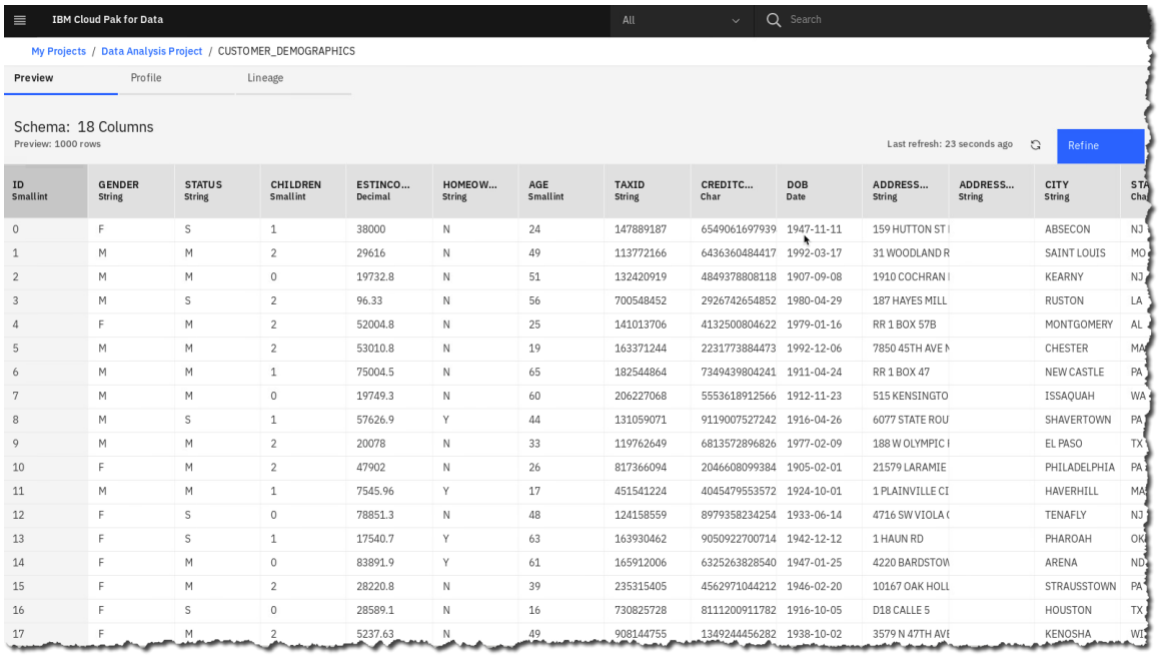
13.5.3 Exploring and Profiling data

One of the most powerful capabilities of Cloud Pak for Data is its ability to automatically create descriptive information, business classifications, etc., about your data. This information will then be exploited to do things such as determine where sensitive information is, decide how we should handle such information, describe who can see it and who cannot, assign business terminology to the data to make it easy for business users to locate and understand, and a whole host of other capabilities.

__33. Next we'll start to get an understanding of our data; click on the CUSTOMER_DEMOGRAPHICS table.

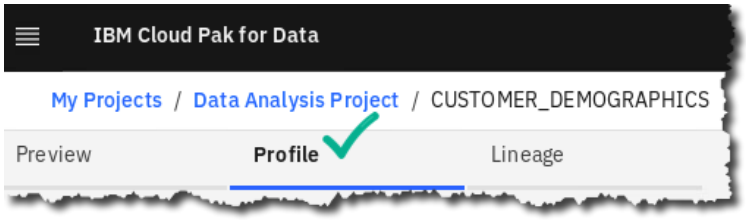


__34. This will bring up a preview window showing you a sample of our data, and a number of options to work with it. Take a minute to familiarize yourself with this screen.



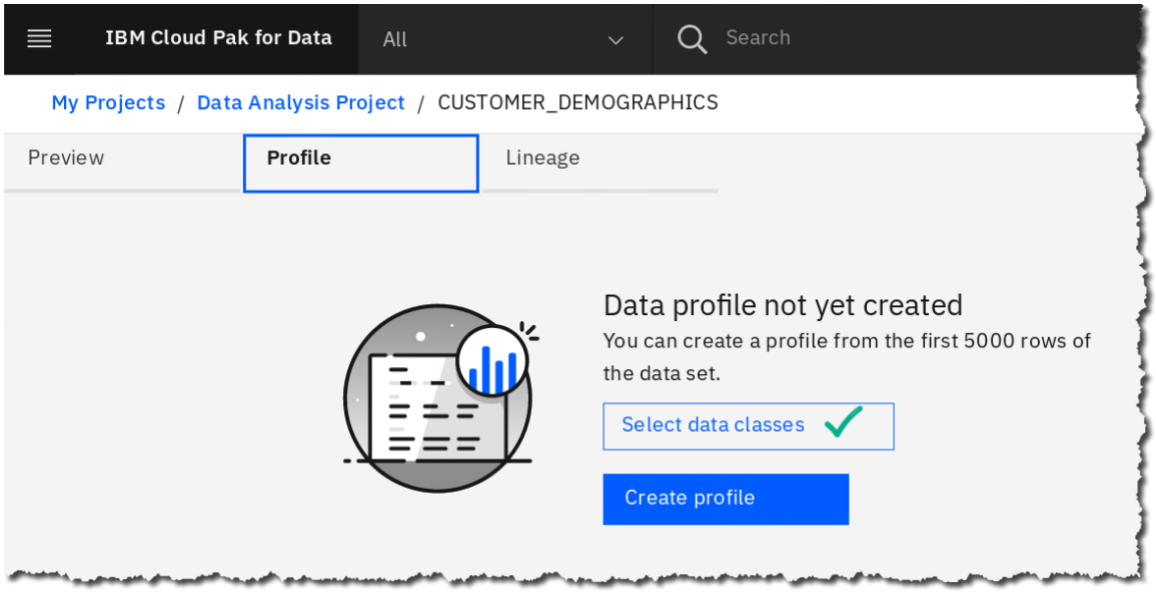
__35. The first thing we'll do is profile the data; profiling is a common practice of inspecting the data to understand basic characteristics of your columns such as frequency distributions, formats, completeness, etc. In Cloud Pak for Data, you will see that it goes a lot further than that…

__36. Click the tab Profile.



This will open the profile launch window.

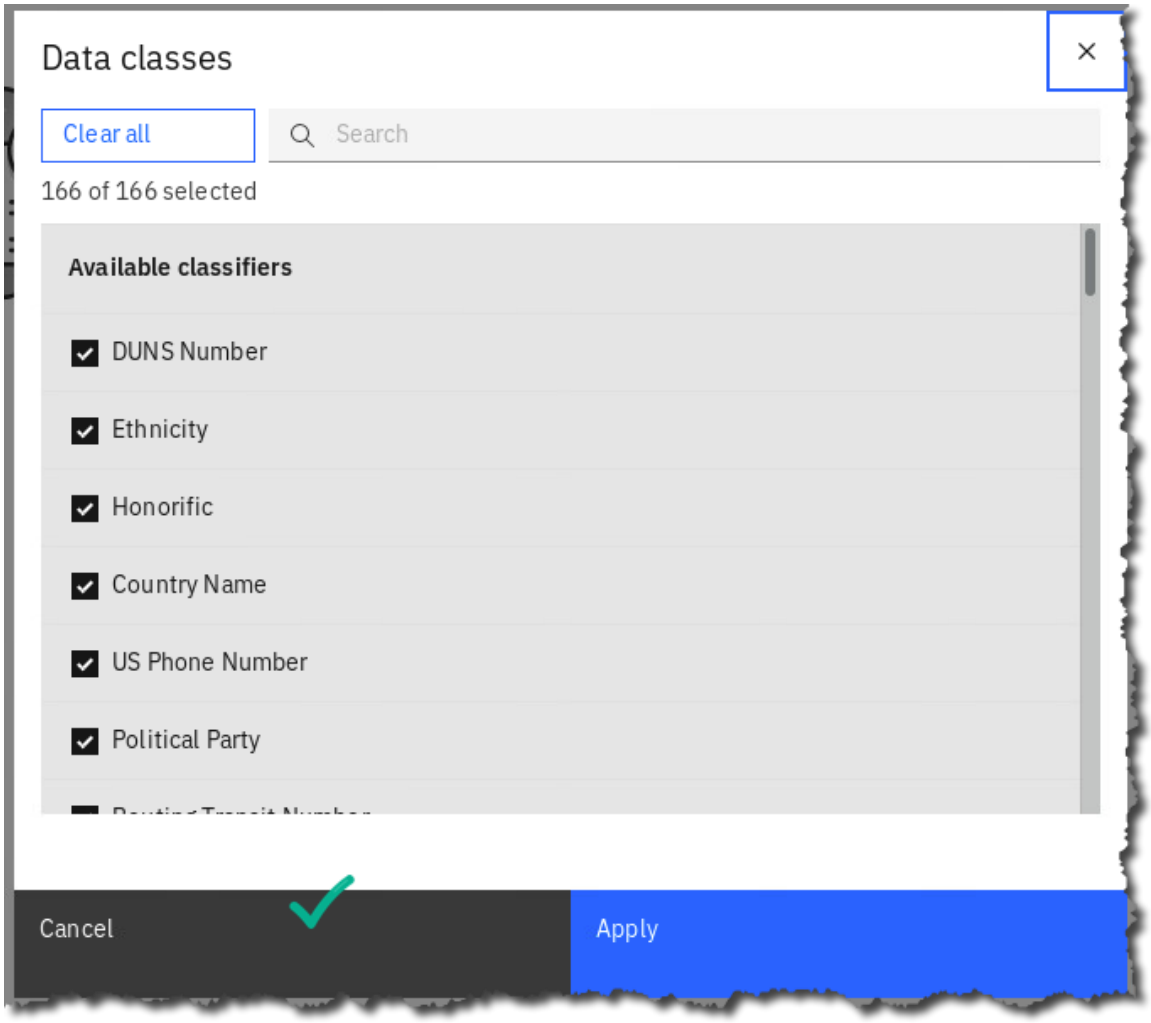
__37. First, click Select Data Classes.



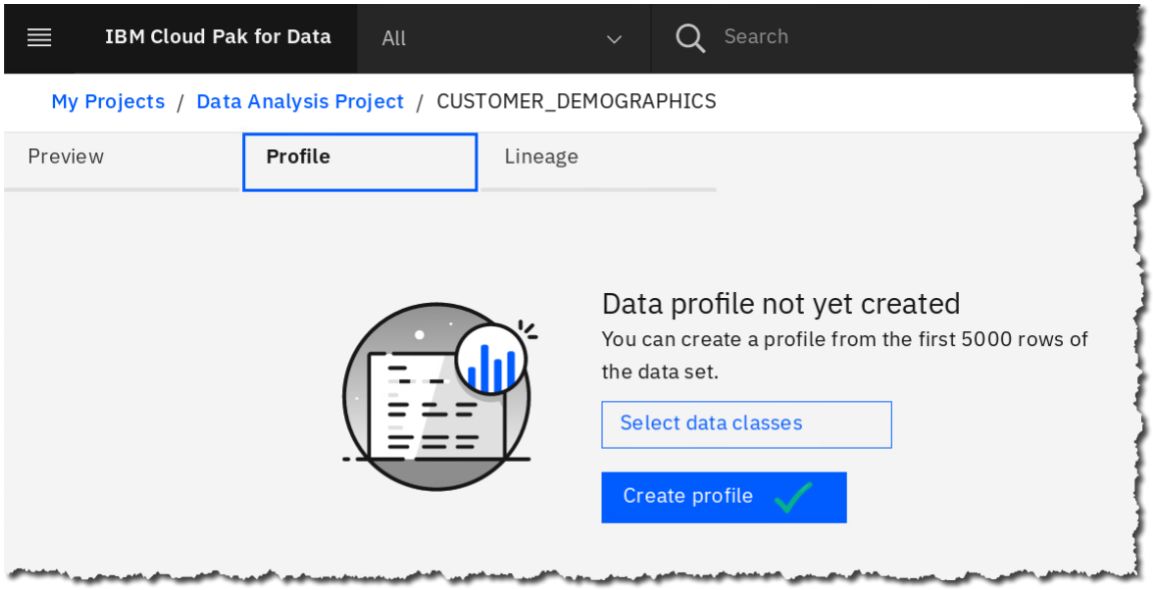
__38. This will show a list of all “out of the box’ data classes that can be assigned automatically during profiling to the different data columns. (Sometimes this does not populate – if so, Cancel and reselect.)

This window allows you to de-select certain classes if they aren’t pertinent to your business or to this profiling exercise. For now, we will leave them all active. In addition, custom data classes can be defined for your business and added to this repository.

Click Cancel (to return to the profile launch screen).

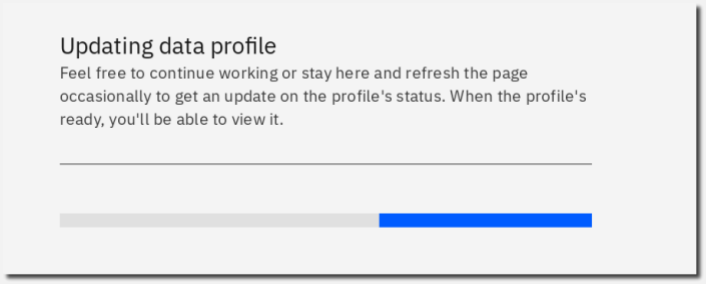


__39. Now click Create Profile.

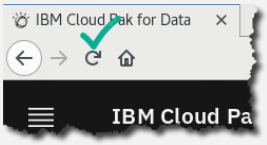


Data Steward

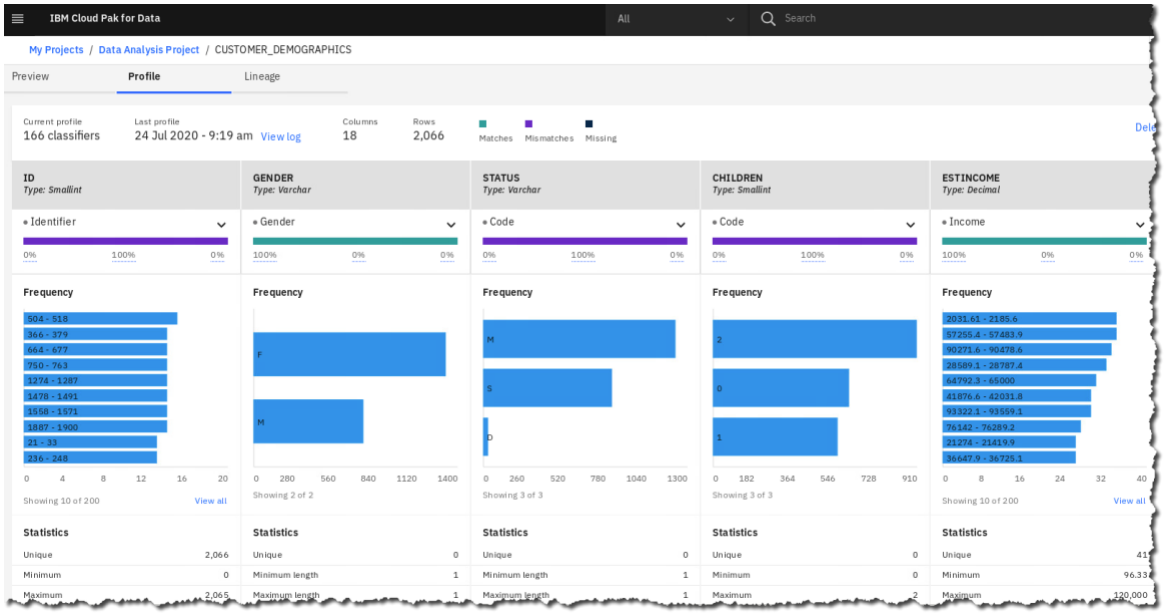
Note: The profiling process will launch. This will take a few moments to complete so feel free to leave this screen and return, but it should only take between 2-3 minutes.



Click refresh on the browser to see if it finished.

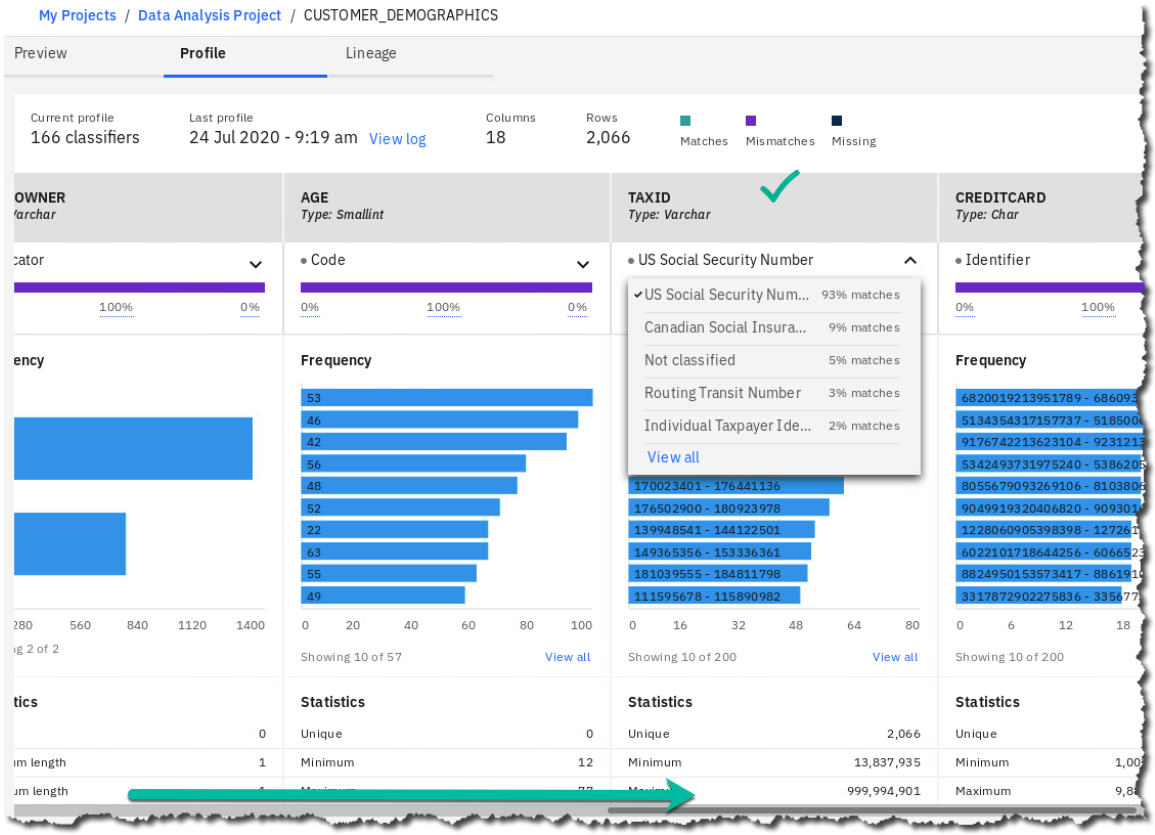


__40. When it is complete, you will see:

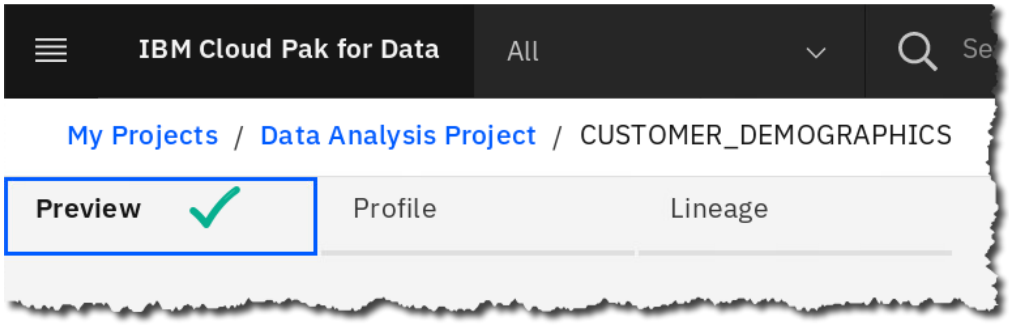


__41. This is a bird's eye view of the results that have been generated by the profile run. Note that you can immediately determine frequencies and pay attention to the data classes that have been identified. Here we are looking at Gender, which was automatically determined during profiling based on the column metadata and the data itself.

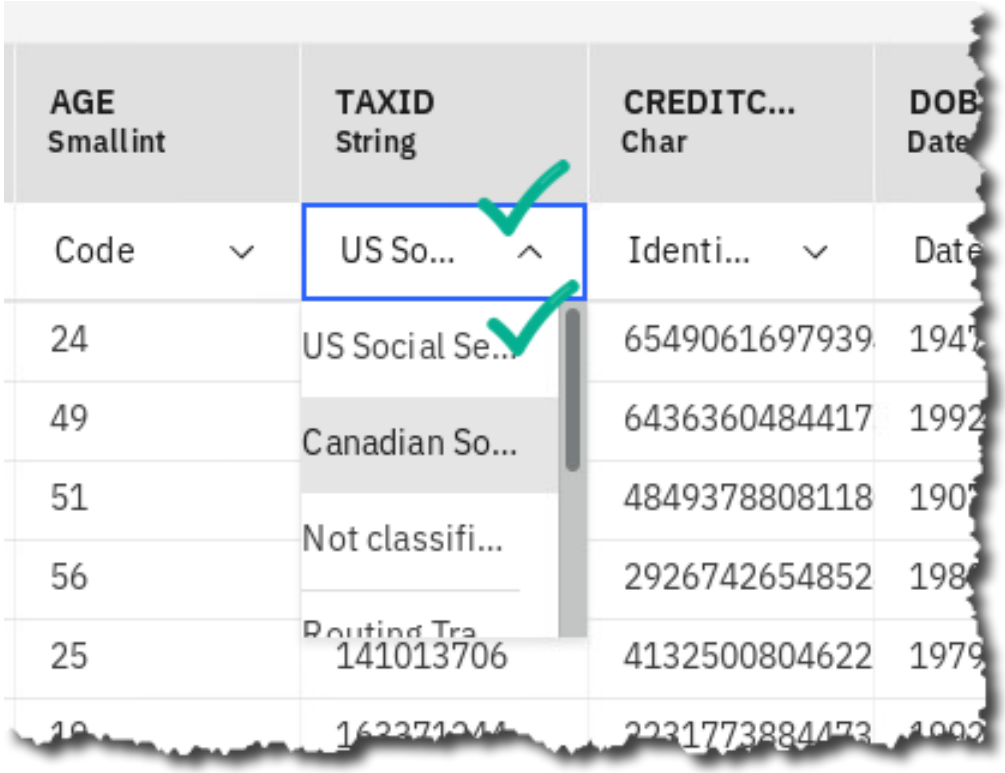
Scroll right to see other columns, like TAXID…



__42. Click back to the section Preview.



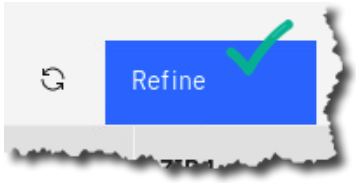
__43. Click on the down arrow for TAXID to see how Profiling found the identifier: US Social Security Number.



13.5.4 Refining visualizations

The next steps will allow us to use visualization techniques to further get an understanding of our data; visualization is a powerful way to get quick insights and make decisions about what we want to do next.

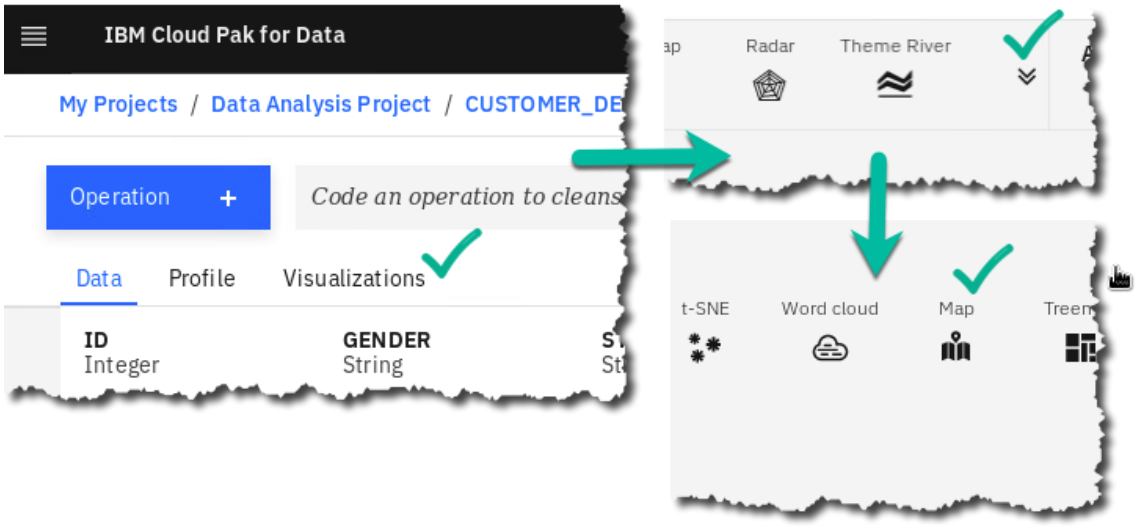
__44. Click Refine.



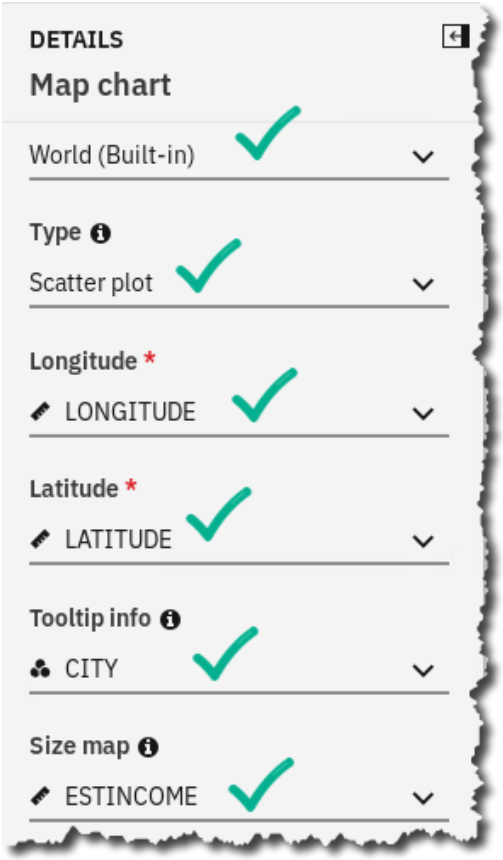
Data Steward	Note: If you receive this error, back up in the browser and refresh to fix. 

__45. Click Visualizations ⇨ Map.

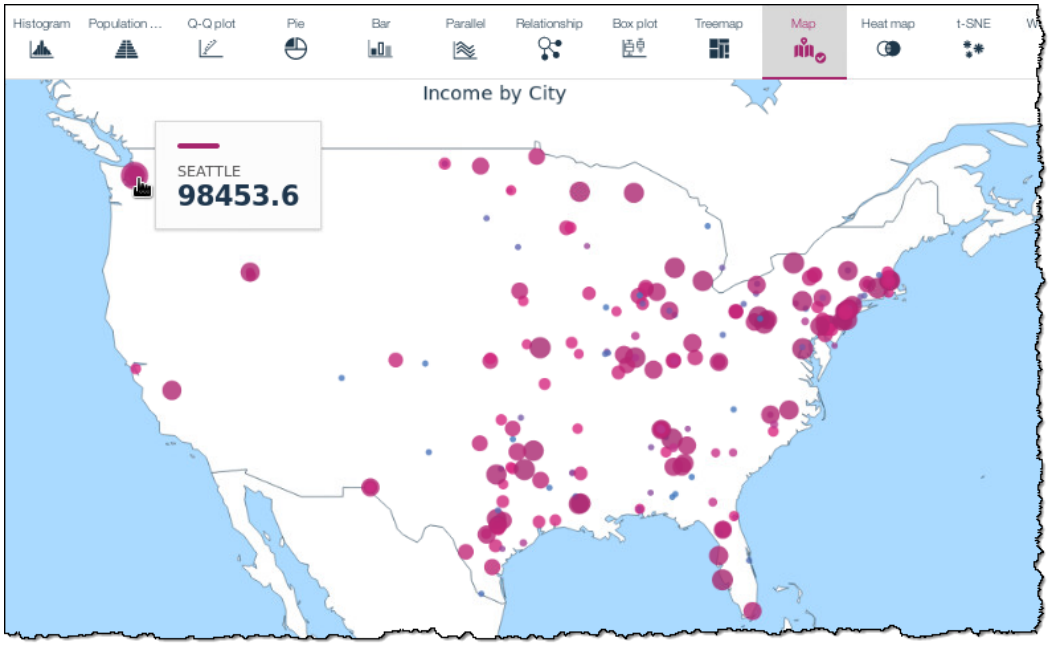
NOTE: You may need to click the double drop down arrow to expand the list of visualizations to find Map.



__46. Fill in the Details as shown below.

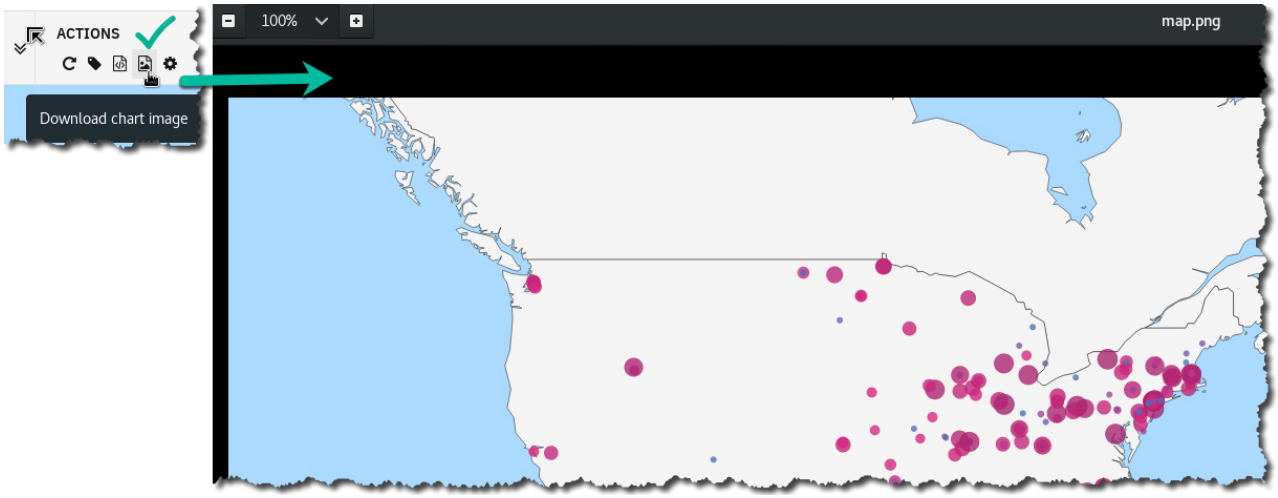


__47. Use your mouse's wheel to zoom in and center the United States on the visualization. Hover over any circle on the map, which indicates the Estimated Income by that City.

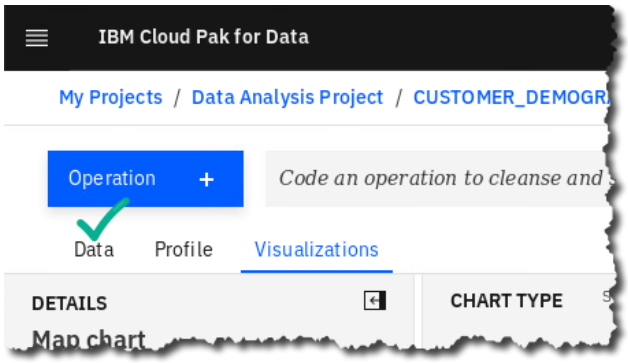


__48. If the Data Scientist or Business Analyst want to keep a visualization, they can download it by clicking on the image icon shown.

Then, they can either save it, or review it with Open with Image Viewer (depending on browser and OS).



__49. Close the Image viewer (if you have it open) and click on the Data tab to return to your data view (you may see a popup asking if you want to leave the page after clicking Data. If so, choose yes).

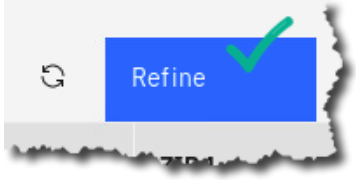


13.5.5 Data refinery flows

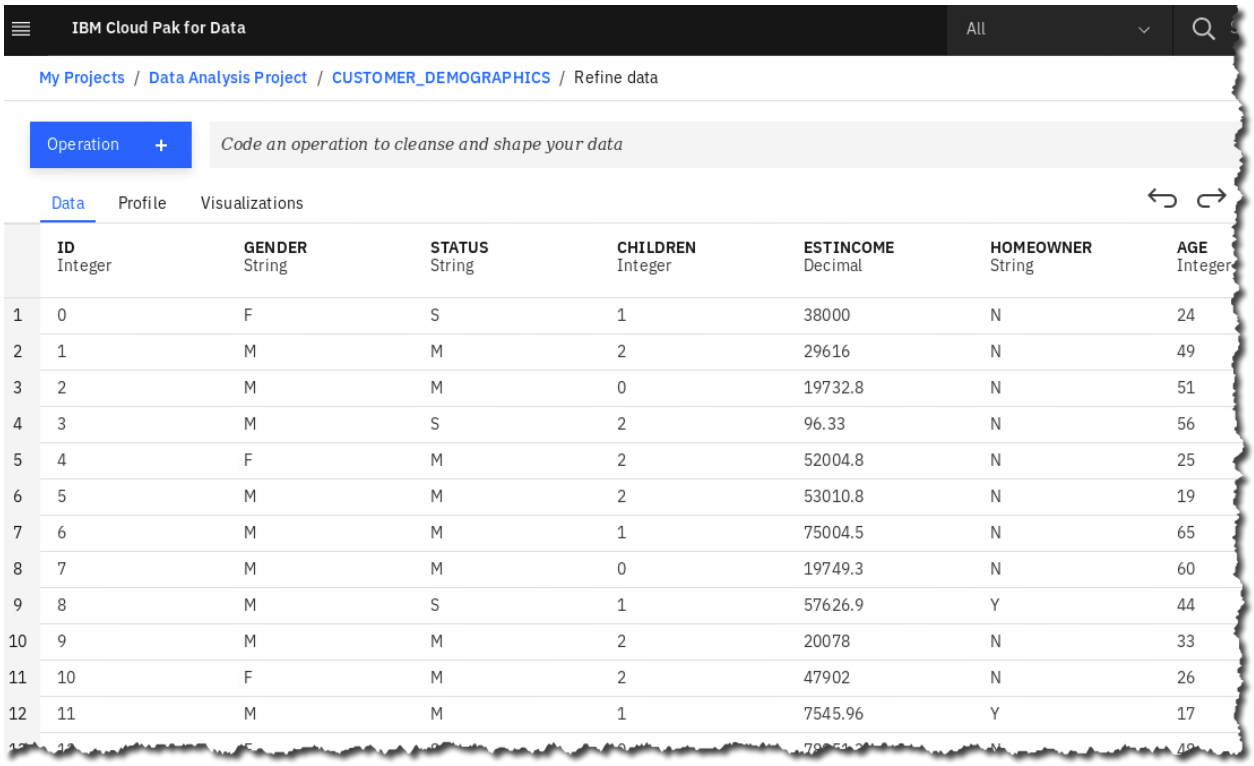
You should now be back at the data grid view page. (See below)

Data Refinery flows are a powerful way to make adjustments to your data, in a visual, self-service environment. As you choose steps, it will create a pipeline process; each step is reversible and editable as well so that you may back up in case of error.

__50. Click Refine if you are not already in the Data Refinery Flow Editor.



__51. The screen will refresh with a slightly different look; you'll still have your data grid, but you'll also see some new options for 'Operation', 'Steps', etc.



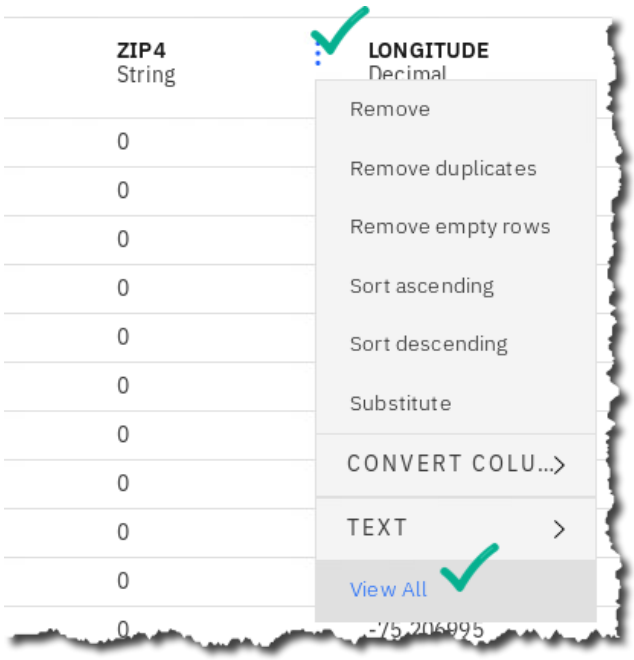
	ID Integer	GENDER String	STATUS String	CHILDREN Integer	ESTINCOME Decimal	HOMEOWNER String	AGE Integer
1	0	F	S	1	38000	N	24
2	1	M	M	2	29616	N	49
3	2	M	M	0	19732.8	N	51
4	3	M	S	2	96.33	N	56
5	4	F	M	2	52004.8	N	25
6	5	M	M	2	53010.8	N	19
7	6	M	M	1	75004.5	N	65
8	7	M	M	0	19749.3	N	60
9	8	M	S	1	57626.9	Y	44
10	9	M	M	2	20078	N	33
11	10	F	M	2	47902	N	26
12	11	M	M	1	7545.96	Y	17

__52. You have been working in what is known as the Data Refinery. This module gives you many different capabilities for transforming, or "shaping" your data, using an easy to understand spreadsheet and menu-driven function paradigm.

The first thing we want to do is scroll over to our two zip code fields, ZIP and ZIP4. Note that the zip field sometimes contains only 4 digits instead of 5. This commonly happens when the field at some point was defined as numeric and the leading digit is a zero – it automatically gets dropped. Many east coast regions have zip codes that start with zero – we'll want to pad the field to get the 0 back. Secondly, we'll want to pad all ZIP4 fields to have 4 zeros if there is a zero in the field.

__53. Click on the ellipses next to the ZIP4 field. This will bring up a floating menu with a number of actions, or what we call operations, that you can use to transform our zip field. Some commonly used functions are called out directly, but we want to pad, so…

Click on View All.



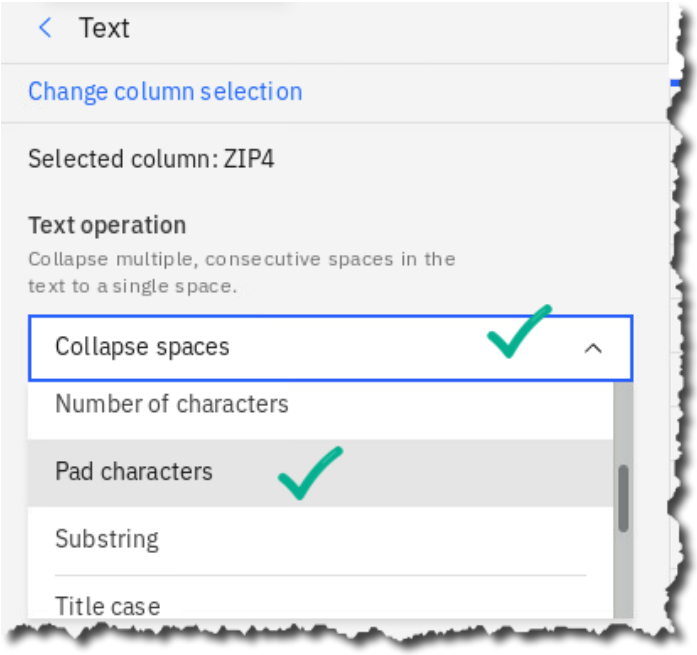
Tip: Use your browser zoom in/out feature to see the View All option.

__54. Now we see all of the shaping operations by category on the left, and the column we're concentrating on, ZIP4.

We're making a text transformation to pad missing numbers with zeros, so click Text.

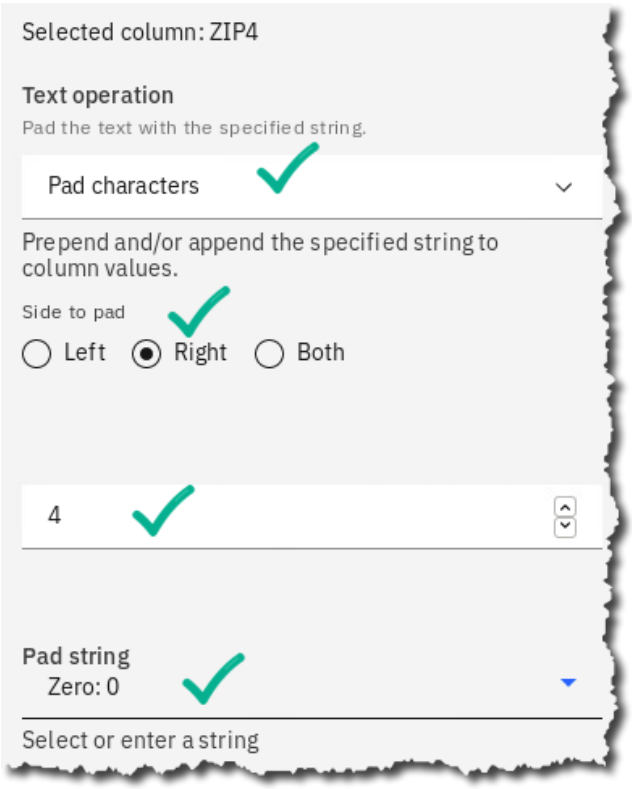
__55. Next, we choose the operation – click the dropdown below Text Operation.

Choose Pad Characters.



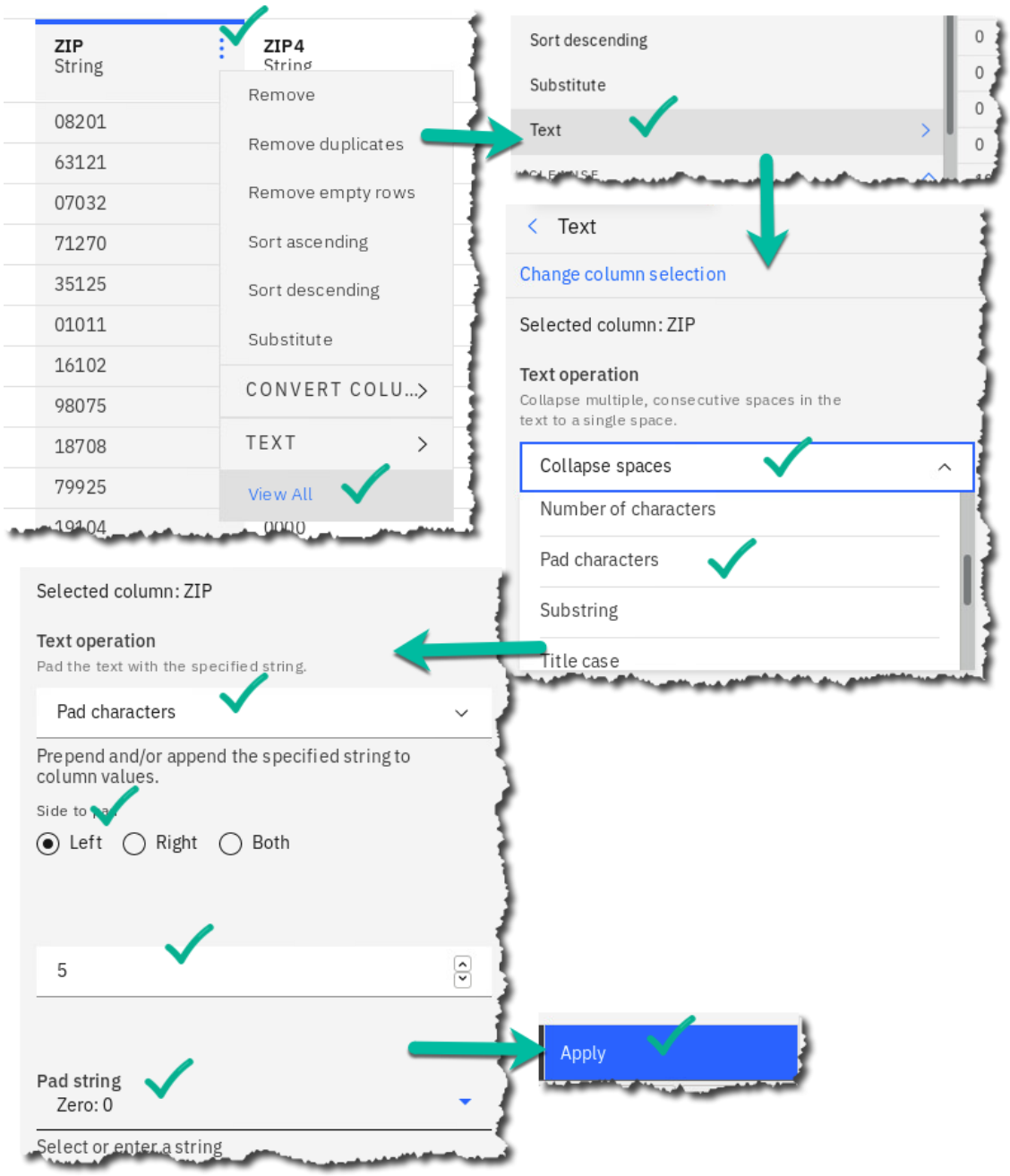
__56. This brings up a window where you put in the parameters for padding characters. This function takes 3 parameters – Side to pad: (Right), what the length of the resulting field should be (4), and what pad string to use: (0).

Here are the parameters you should choose:



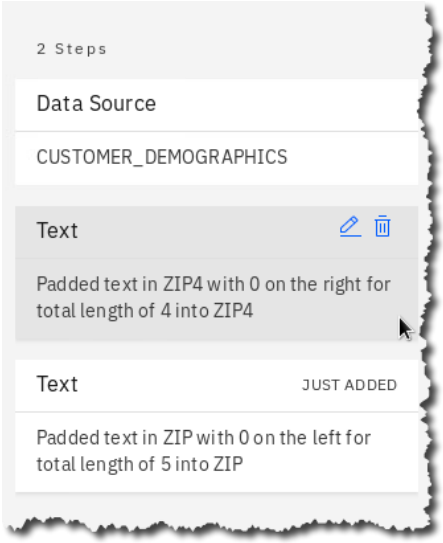
__57. After entering these parameters, click the Apply button. This will take you back to the data grid, and our field ZIP4 will now be padded if necessary.

__58. We're now going to do a similar thing to the ZIP field. We want to pad the zip field with a zero, if necessary, to make the field 5 characters. *Don't look at the screenshot below!* Try it on your own first – but if you're struggling, the screenshots below should help.



__59. When you click Apply you're returned to the data grid. Scroll over to ZIP and verify the leading zeros are in place:

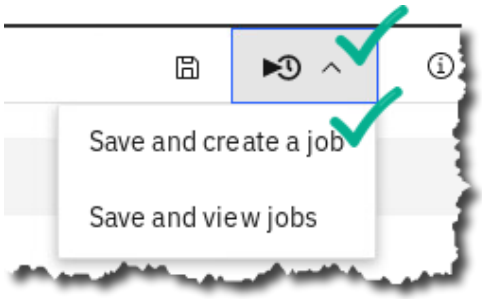
__60. What's also great about refinery is that as you add operations it keeps a list of those operations, in easy to understand English, not code – and you can delete a step at any point by clicking the garbage can on that step – the data will revert back to its state prior to the step.



__61. Now that we've built a pipeline with two steps, we want to run it so we can persist the data somewhere. To do that, we save the pipeline as a "job".

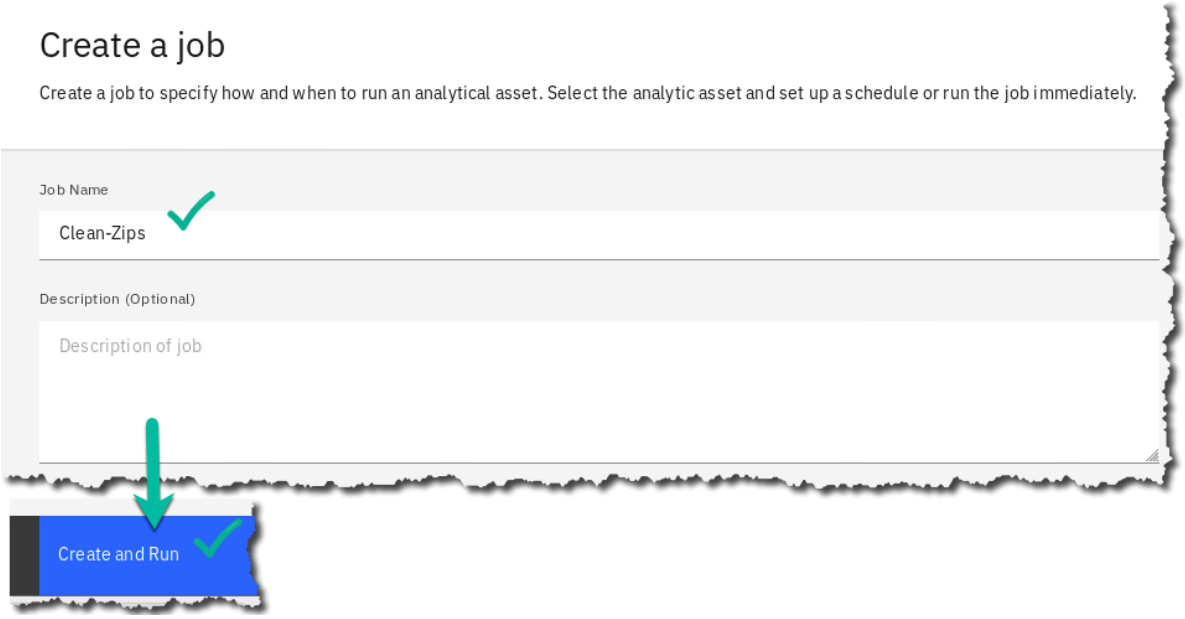
Click the down arrow icon next to the clock and play arrow icons,

then click Save and Create a Job.



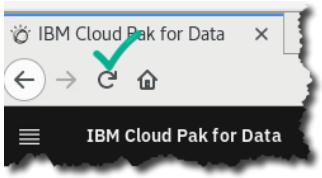
__62. Next we need to name the job and then run it.

Enter the name Clean-Zips ⇨ Create and Run.



__63. The job takes about a minute to complete. You can see it running on the screen.

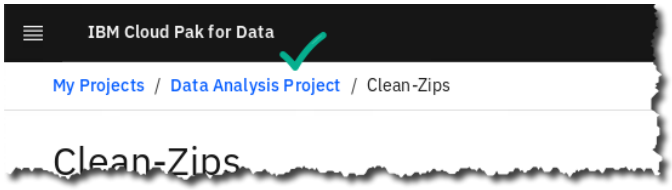
Hit the refresh icon to see the status more quickly.



Data Steward	Note: Due to an issue, the first attempt to run the job may fail; if so, just re-run and it should be successful.

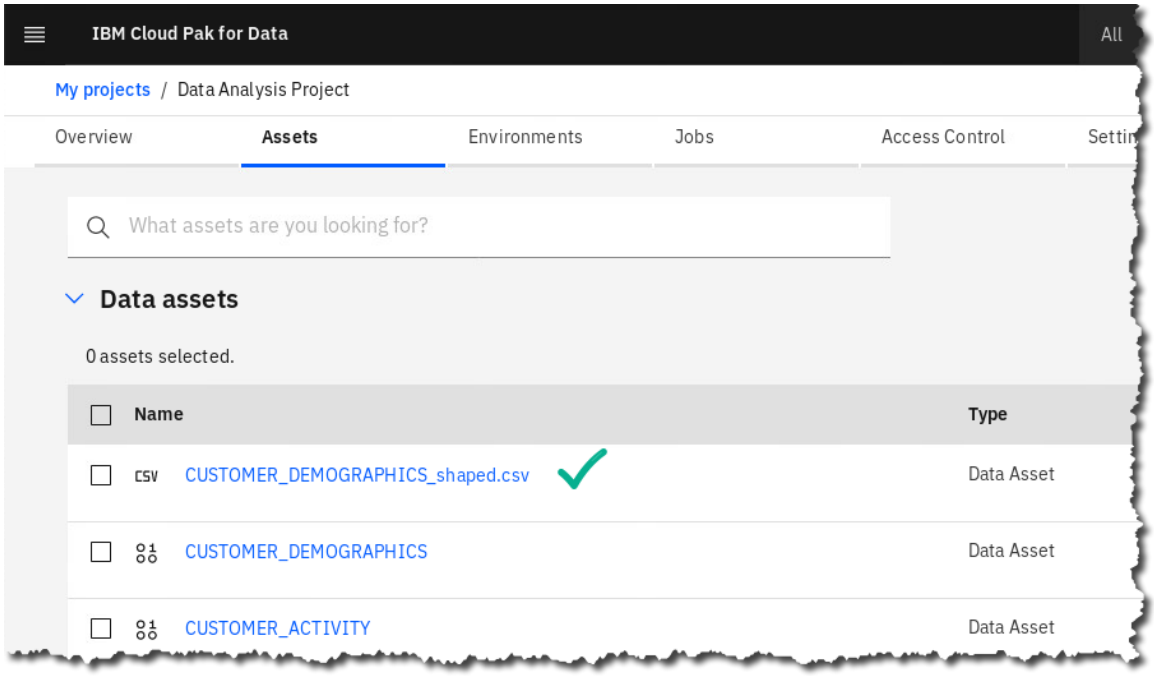
__64. When Completed, you can check the logs of the job if you wanted to. This can be especially useful if the job is complex. This job is only a simple two step job so we will not review it now.

__65. Click on the breadcrumb trail to return back to the project.

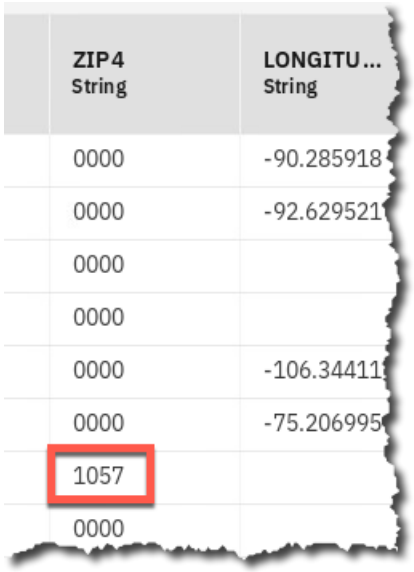


__66. A new output was created from the Refine job.

Click on it: CUSTOMER_DEMOGRAPHICS_shaped.csv.



__67. Scroll over to find the ZIP and ZIP4 columns. Notice the data is padded with four zeros when there was previously only one zero in ZIP4 and the ZIP field is left padded with a 0.



Data Steward	The power of Refine can be used by more than just the Data Steward because it is launched from a Project, which enables many more personas the ability to shape and refine data assets that the Data Steward may have given them access to. This gives anyone the ability to explore and shape data on their own with self-service capabilities. If you need to make a permanent change to the data, for example, to write it back to a database from where it came, that is where Transform comes in. We will explore Transform later in this lab.

13.6 Reviewing the business glossary

Business Glossaries have been around in various incarnations for about ten years. They have enjoyed varying degrees of success. The problem with glossaries, and with data governance in general, is that it's too costly and manually-intensive to maintain and enhance these glossaries, and the associated metadata, etc., over a long period of time. Today you'll see how Cloud Pak for Data's Automation capabilities finally make well-maintained, value-added governance solutions a reality.

A business glossary consists of categories and terms.

Categories provide the logical structure for the glossary so that you can browse and understand the relationships among terms and categories in the glossary. Categories can be organized in a hierarchy based on their meaning and relationships to one another.

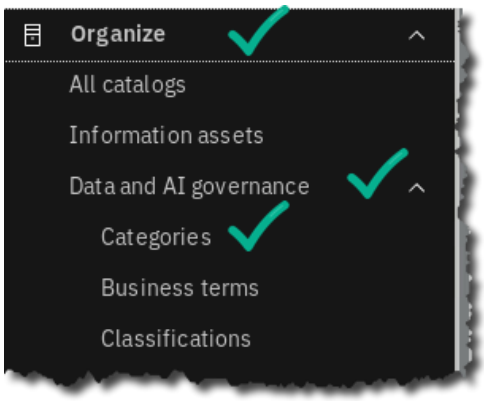
A Business Term is a word or phrase that describes a characteristic of the enterprise. Terms are the fundamental building blocks of the glossary. Each Business Term has a parent Category, but it can also be referenced by other Categories. When you create a Business Term, you need to provide a meaningful name. Business Terms can be assigned to other Business Terms, and to other asset types as well.

Business Terms can be associated with a variety of different data assets in your organization, from tables and columns and files and fields to policies, ETL processes, data models, etc. This makes it easy for business users to find what they are looking for using terminology they understand.

Let's explore some of the categories and terms that are pre-loaded into our environment.

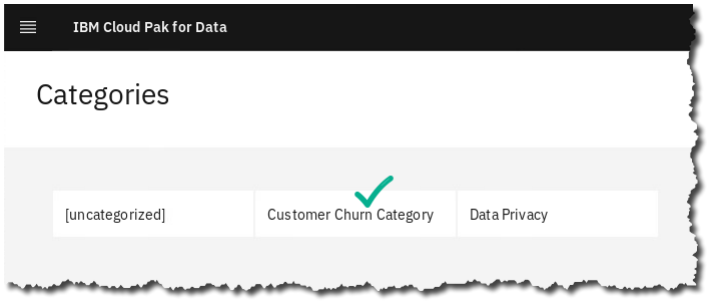
__68. Start at the Navigation Menu.

Click Organize ⇨ Data and AI governance ⇨ Categories.

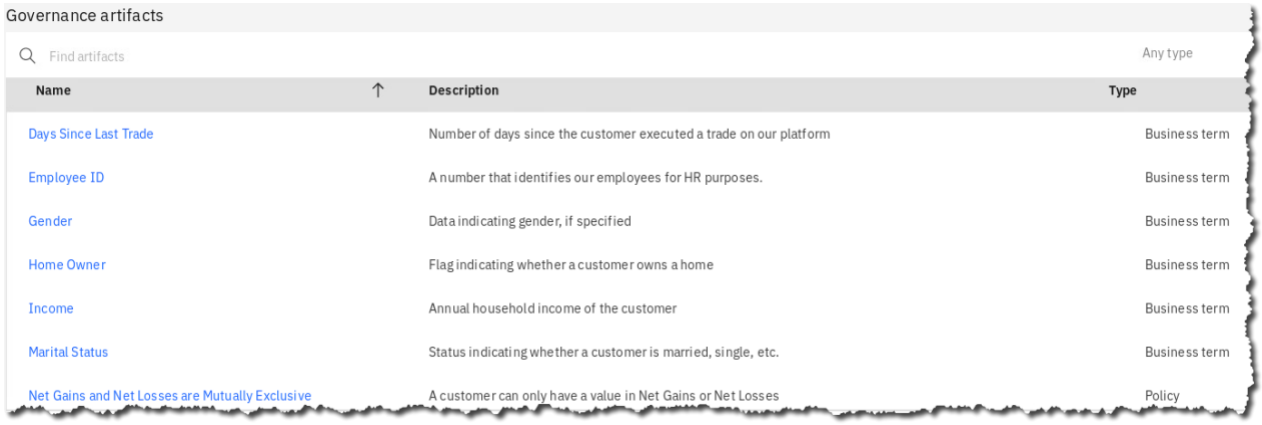


__69. Review a category already created for you to review.

Click Customer Churn Category.

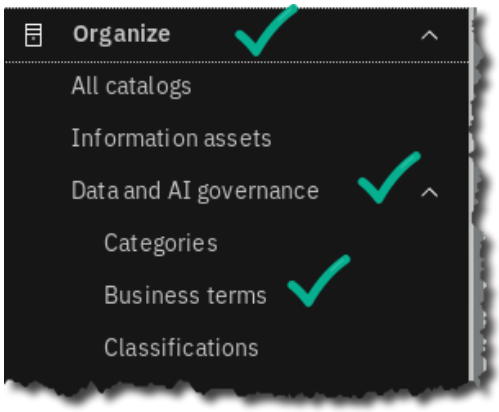


__70. Review the Business Terms and Policy for the Category.

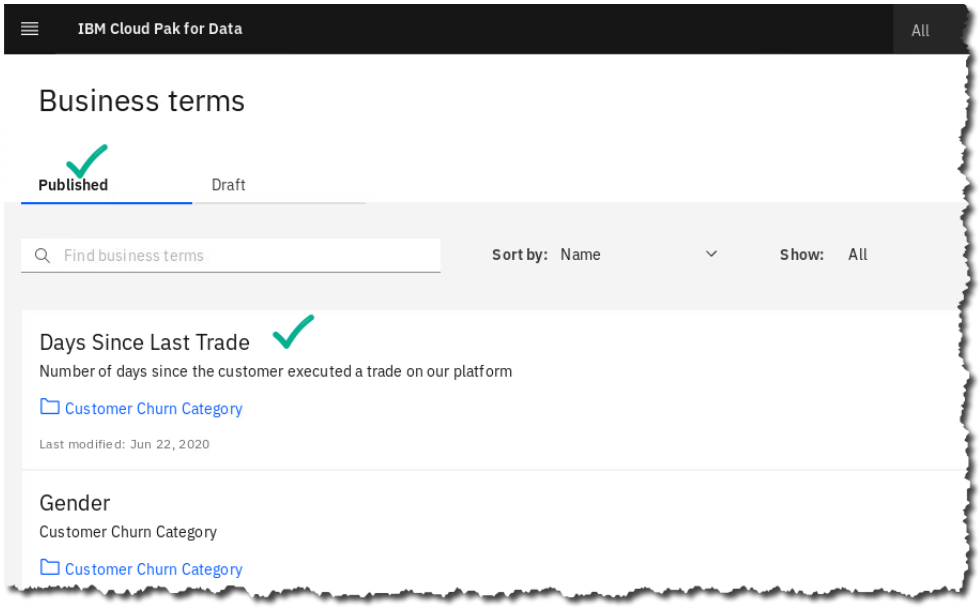


Notice that on this page you could create a new Category, but do not do so. We will only be working with this one category.

__71. Click Navigation Menu ⇨ Organize ⇨ Data and AI governance ⇨ Business terms.



__72. Here you can edit or add new Business terms that are either in Published or Draft mode. Click on one to review it in more detail.



Data Steward

You can create your own Glossary with Categories and Business Terms manually or import them from a file. In addition, you can import Industry Models from IBM for industries such as finance, banking, healthcare, and insurance, and import them into CPD.

See the services screen  then Industry solutions.

13.7 Automation Capabilities – Discovery, Classification, Term Assignment

As mentioned in the business glossary section, the key to successful data governance lies in the use of automation and machine learning techniques to make decisions, create relationships, and generally enhance the value, timeliness and accuracy of your data governance activities. What you'll be doing next is known as Auto Discovery. This is a process that will crawl the data you point it at, and do several things:

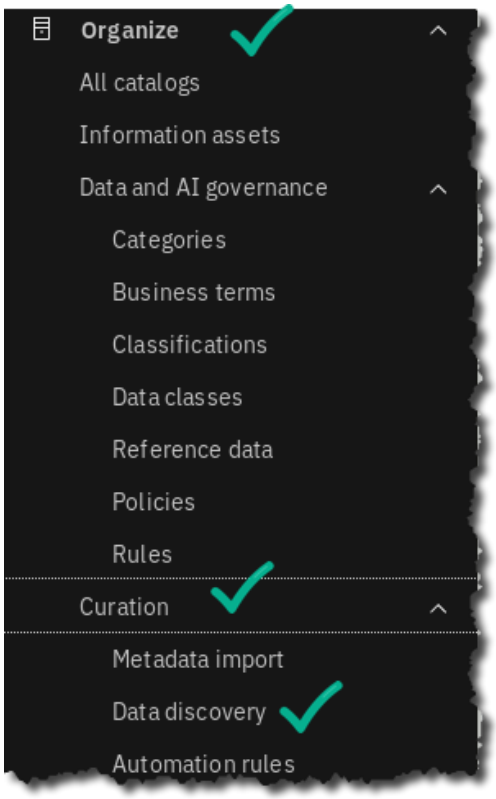
__a. *Classify the data* – this is the process that will assign business classifications to your data elements.

__b. *Score the data* – this will allow you to quickly understand how clean your data is.

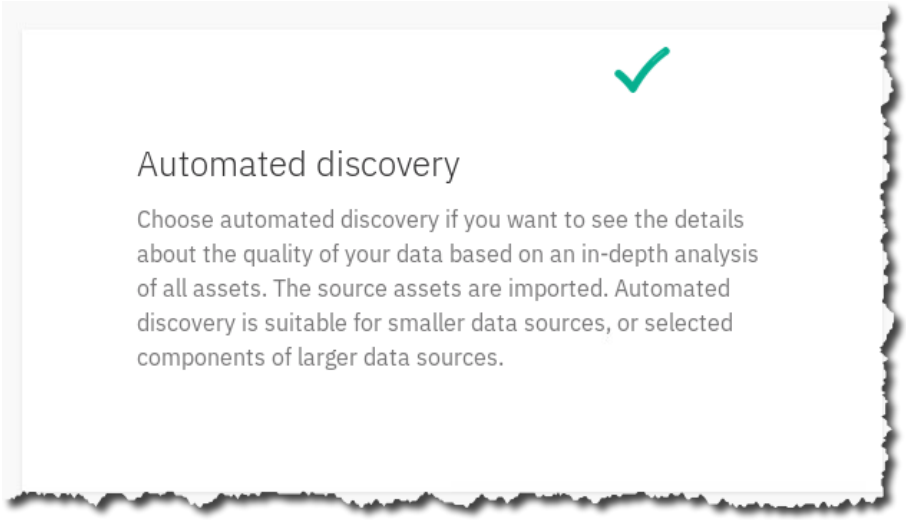
__c. *Assign Terms to your data* – this is where CPD will look at the data you're pointing to, take the business glossary terms, and based on both the data in the columns and the metadata and classifications already determined, assign a confidence rating to specific terms that it matches to the columns. You can set a threshold for when it automatically makes the connection between terms and columns, or you can use a workflow-driven review process to inspect the confidences and determine whether to assign or not.

__73. Let's get started. Go to the Navigation Menu.

Click Organize ⇨ Curation ⇨ Data discovery.

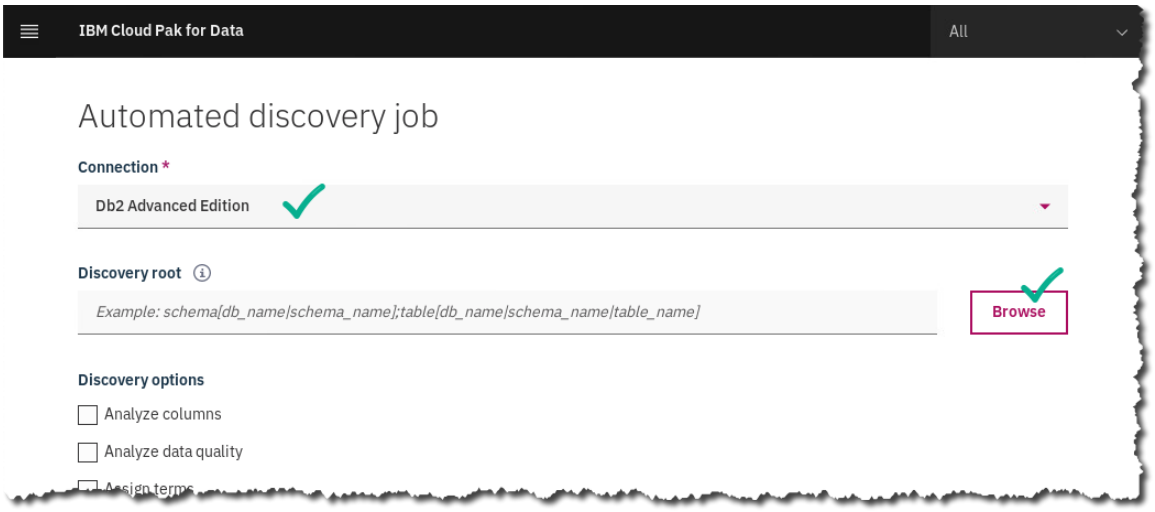


__74. This will bring up a screen showing any past results and allow you to create a new discovery job. Click Automated discovery.



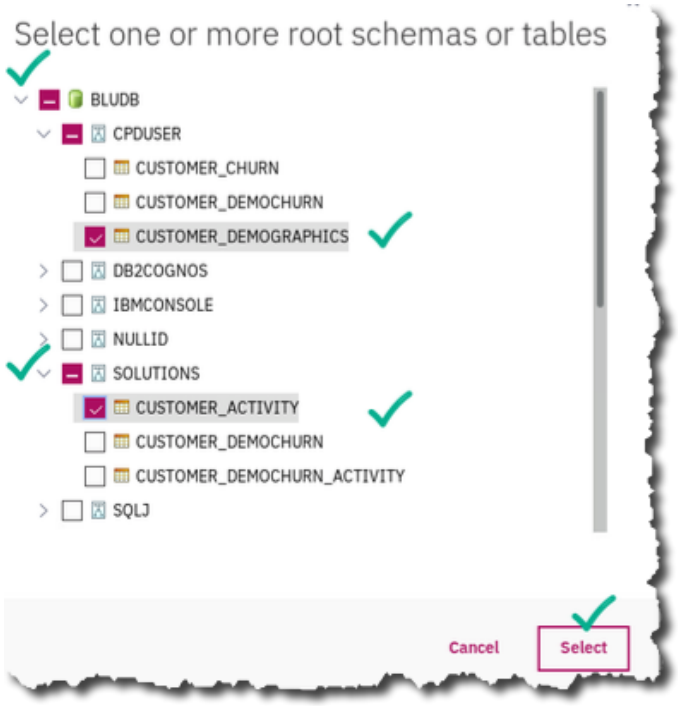
__75. Next, we'll choose the connection that gets us to our data, and the options we want for this discovery run:

For Connection choose the connection we created earlier, i.e., Db2 Advanced Edition, then click Browse.

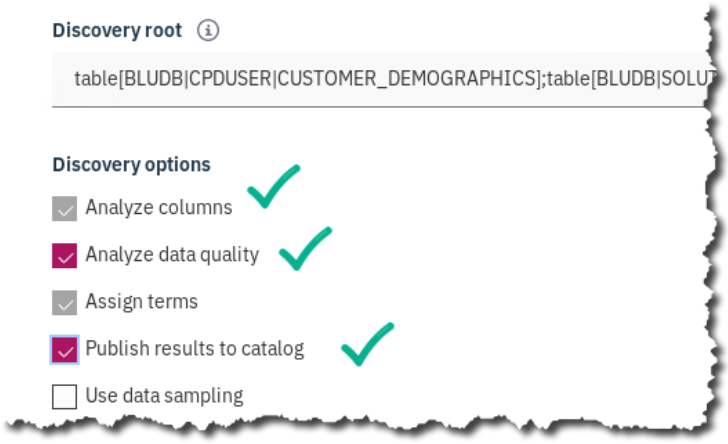


__76. This will bring up a screen where we can choose which tables, schemas, or an entire database for which we want to run Discovery. We're going to choose two tables that will be our main tables for later transformation.

Click to expand BLUDB then CPDUSER ⇨ CUSTOMER_DEMOGRAPHICS, and SOLUTIONS ⇨ CUSTOMER_ACTIVITY. Then Select .



__77. Next, click the checkboxes for Analyze Columns, Analyze Data Quality and Publish to Catalog. (Assign terms will automatically be checked.)

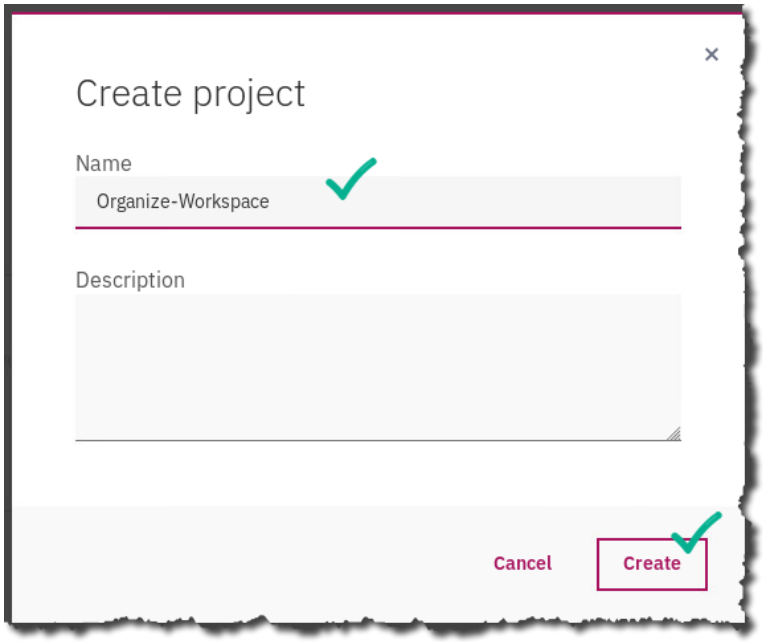


__78. Lastly, you need to choose where to place the results for further analysis or inspection – these are called *Projects.*

Scroll down to find the Select a project dropdown.

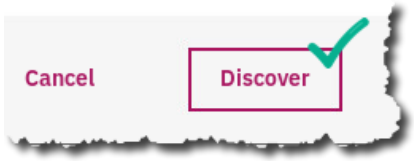
Choose Add a project.

__79. This will bring up a screen where you can name the Project Organize-Workspace and create it.



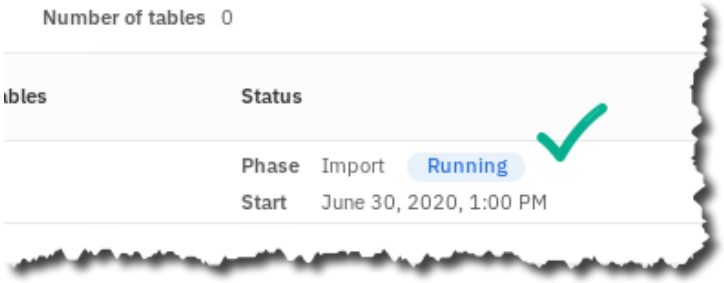
__80. Once the project is created you'll be taken back to the Discovery screen.

Now click Discover.

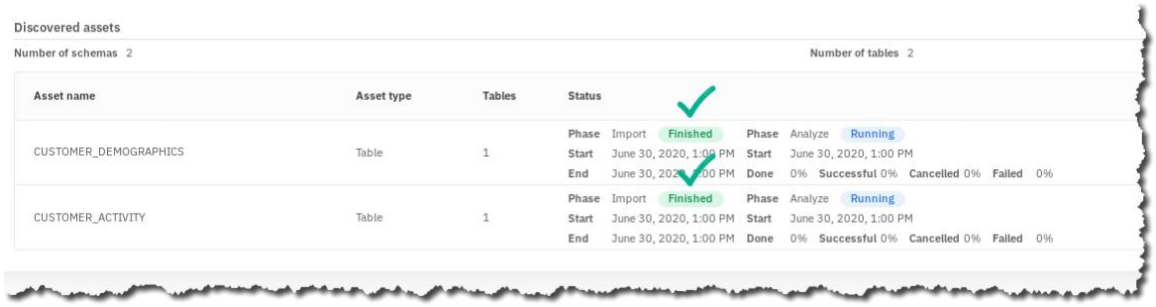


__81. This will bring up a screen where the status of the discovery job is shown; we see the Import portion is running; this imports the metadata from the tables into our workspace.

__82. Click the refresh button on the right to update the status.

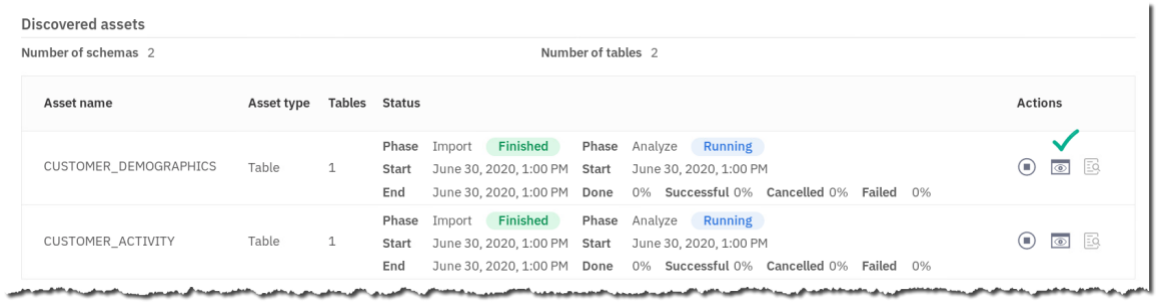


The process will take a minute or two to complete – when it's complete you'll see the blue Running turn to green Finished.



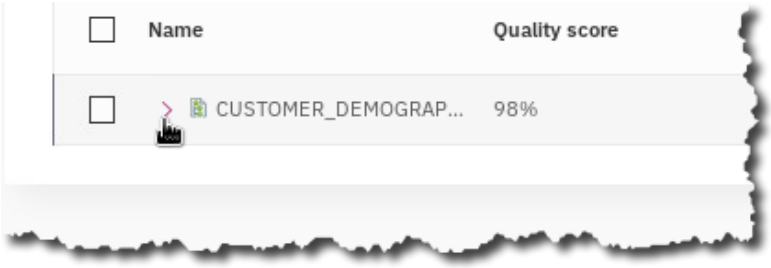
__83. We can now take a look at what the discovery process found.

When finished, click the eye icon next to the CUSTOMER_DEMOGRAPHICS table entry.

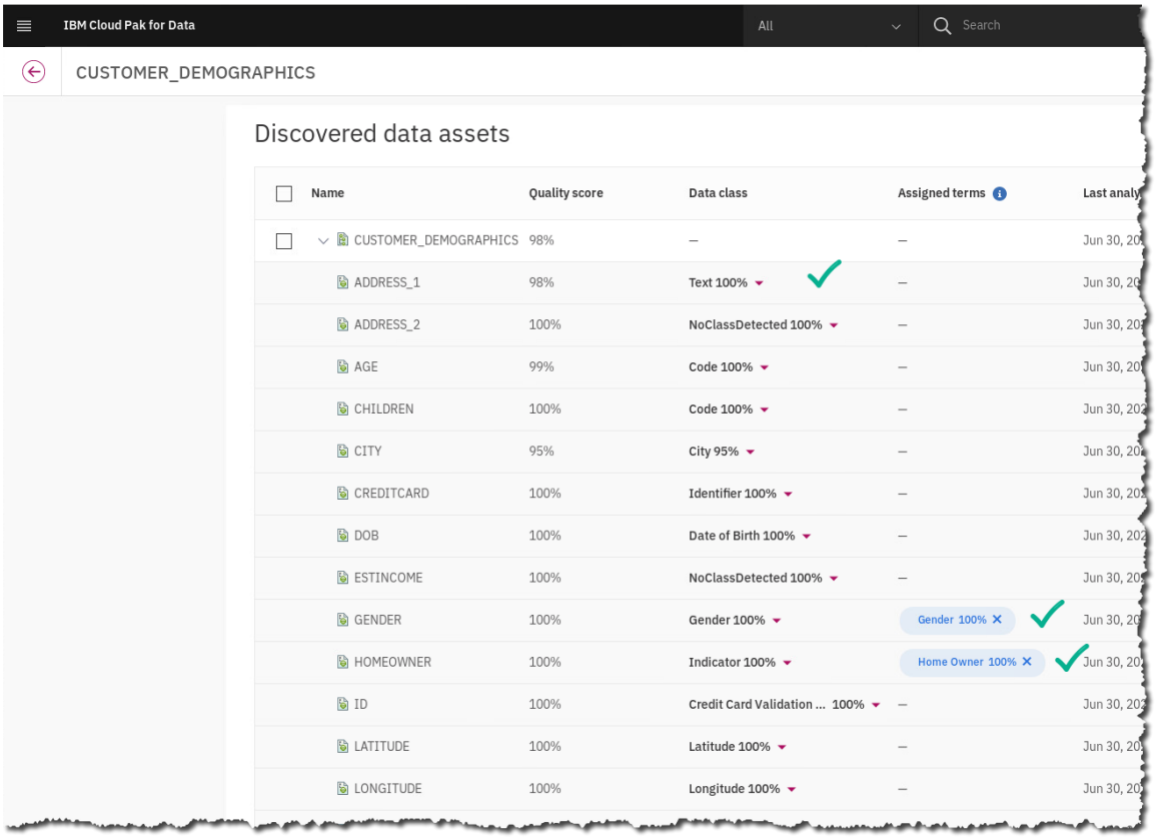


__84. This takes us into the view of our demographics table.

Click the > next to the table name to expand the columns.



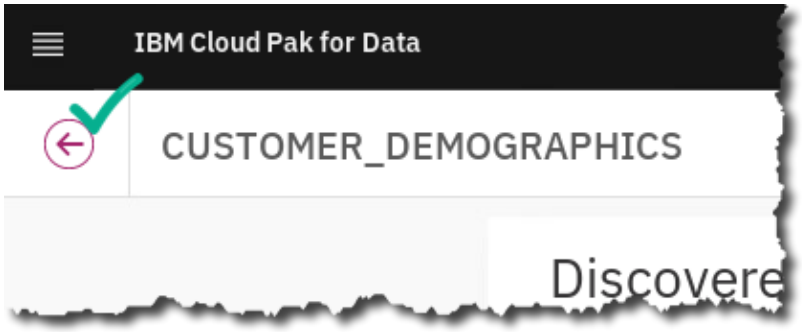
__85. You will then see that the discovery process has automatically determined data classes for many of the columns, as well as assigned Business Terms if the confidence of the term being a match is > 80%. This is incredibly powerful – in the past this sort of work was done completely manually, and often the results were error-ridden. Research suggests this automation eliminates 85%-90% of the time it takes to do this manually.



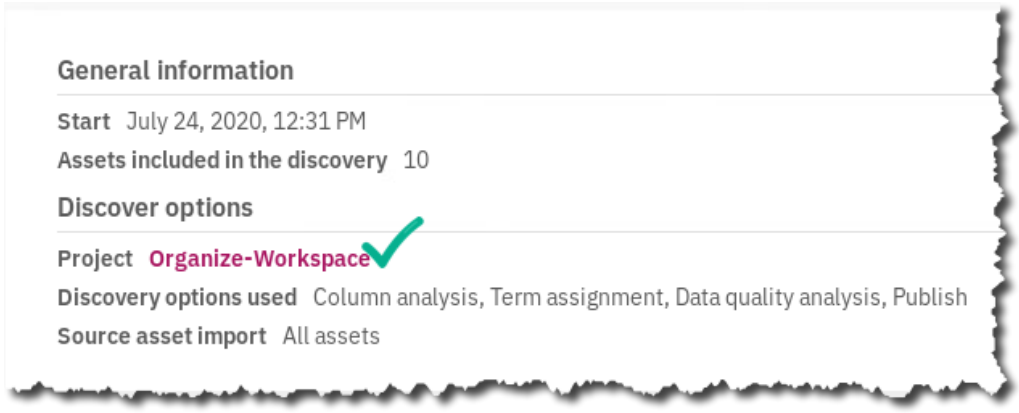
Note: Your Quality score may show numbers slightly different than above.

__86. So, what can we do with this information? Since we decided to automatically publish it to the catalog, we can look at some of the assets and see what we can find. First, let's check the information that's been collected in the workspace from the discovery.

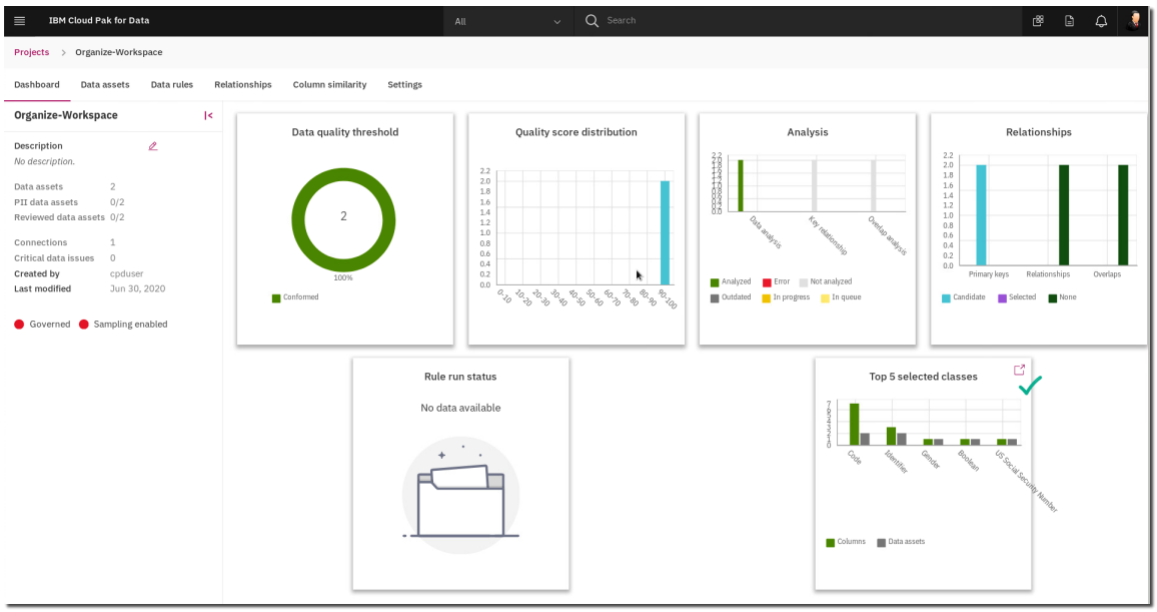
Click the back arrow to return to our main discovery listing.



__87. Next, click the link to jump into the Organize-Workspace project.

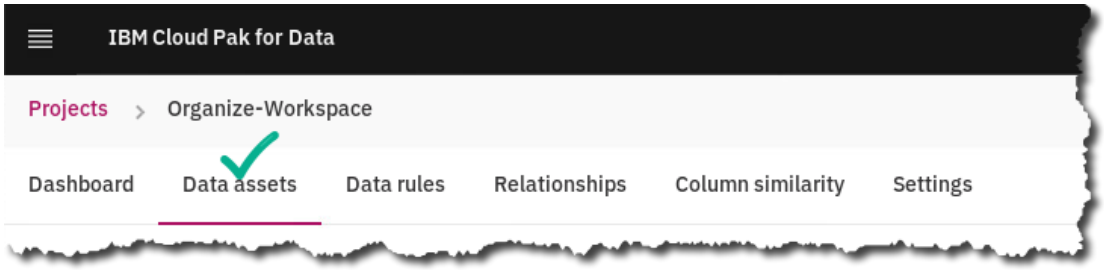


__88. We're now shown a dashboard with some highlighted information about our discovery. If you scroll down, you'll see the top data classes selected from the data that we ran discovery against, and also the quality score distribution across our data sets. The distribution shows that the data sets are pretty clean, scoring 90-100% on quality scores.

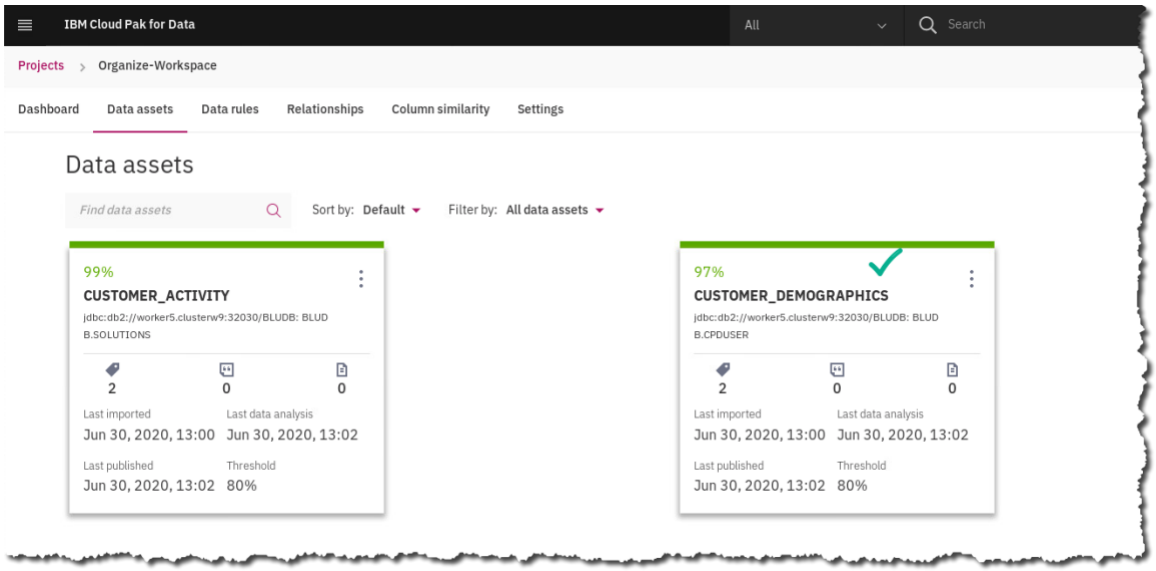


__89. Next, let's look at some of the other characteristics of our data elements that have been discovered.

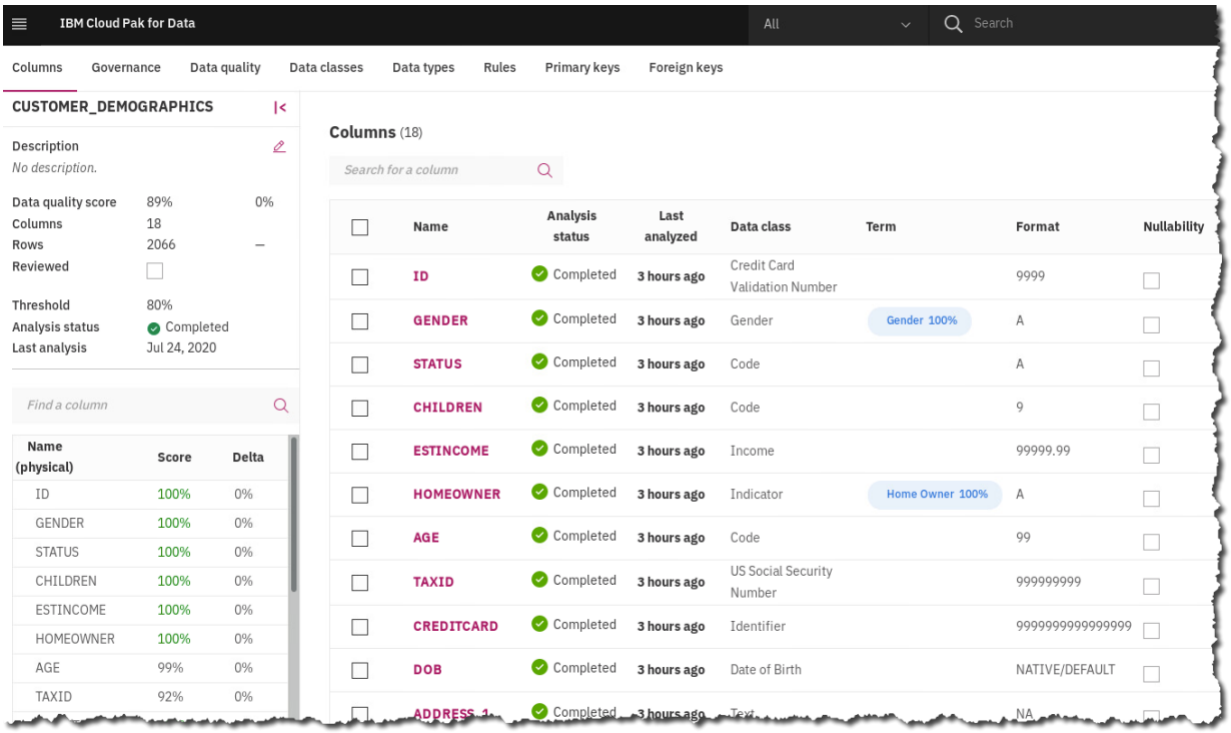
Click the link that says Data assets.



__90. Next, click on the CUSTOMER_DEMOGRAPHICS tile.



__91. We are now looking at all the columns in the table, and different information about each column; Data Class, Business Term assigned, even the prevailing format and whether the field is unique. You can click on a column name to get even more detailed information.



Data Steward

A Data Dictionary contains a Business Glossary (Categories and Business Terms) as well as Information Governance Policies and Rules to ensure data compliance with business objectives, as well as data security.

In this lab we have a beginning sample of these items, but in reality, a Data Dictionary for any organization is quite large and can and should be updated frequently as new data sources, regulations, and other criteria require it.

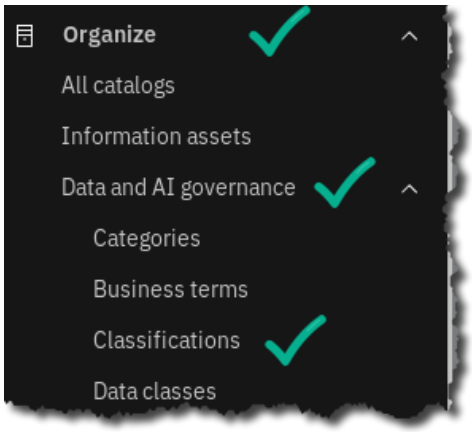
13.8 Reviewing Classifications, Data Classes, and Reference Data

13.8.1 Classifications

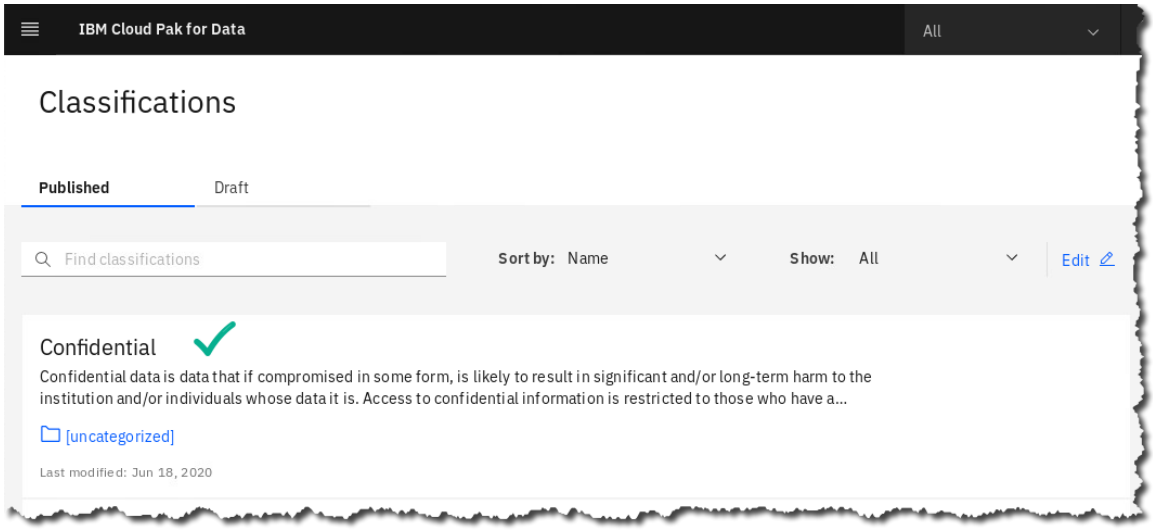
A classification describes the sensitivity level of data. In catalogs, a classification describes the sensitivity of a whole data asset to help catalog members understand the asset. You can use classifications to describe Business Terms, Data Classes, Reference Data Sets, and Governance Rules. In data protection Rules, you can include Classifications in conditions to identify the type of data to restrict.

__92. Start at the Navigation Menu.

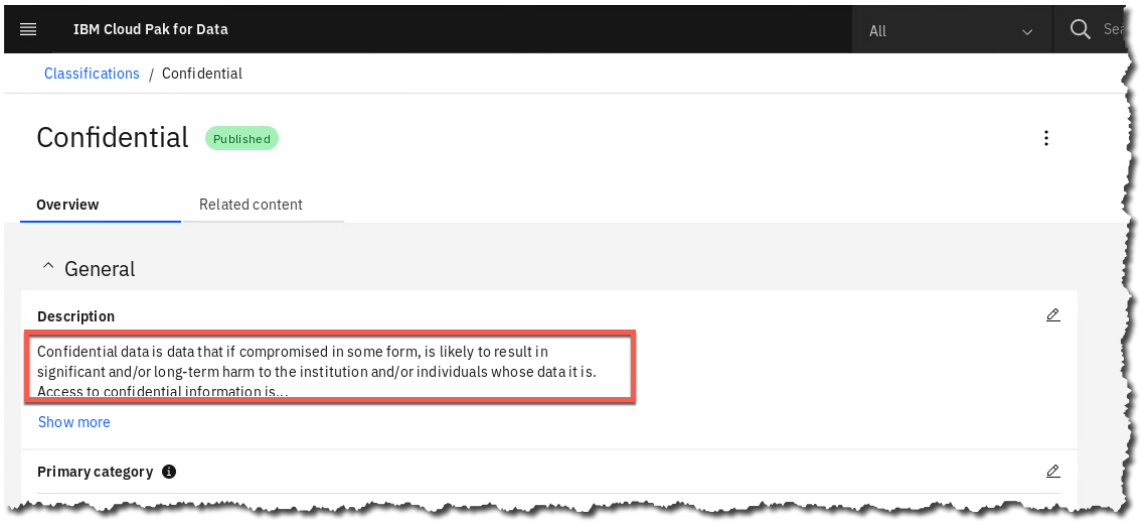
Click Organize ⇨ Data and AI governance ⇨ Classifications.



__93. Scroll to find Classification Confidential and click on it.



__94. The Classification is described here. You could also add the primary Category here, but there is no need to do so now.



13.8.2 Data classes

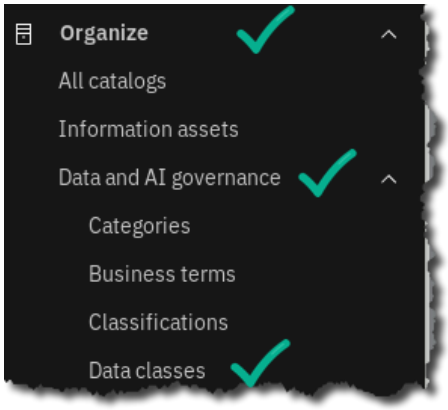
Data classes describe the contents of data in a column of a relational or structured data set. Data classes are assigned to columns when profiling a structured data asset and shown on the **Profile** page in a Catalog or Project.

Watson Knowledge Catalog provides a predefined set of Data Classes. Some Data Classes are categorized into groups, for example:

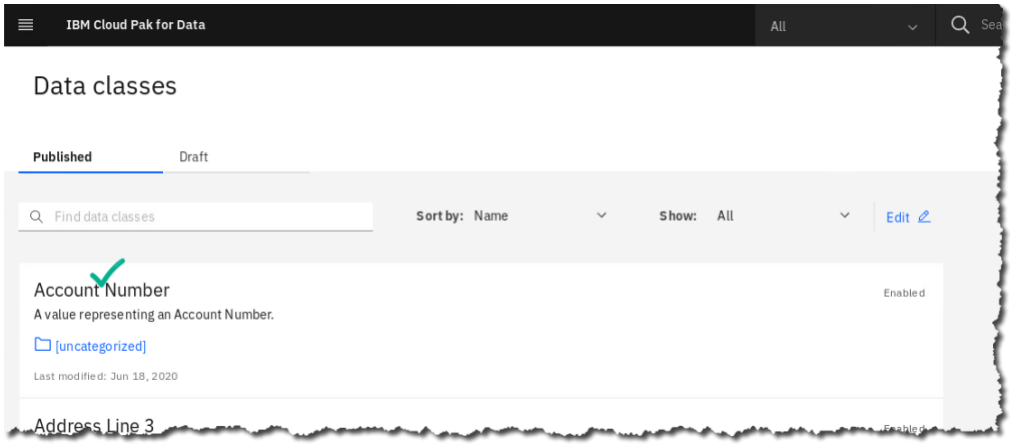
- If you select **Date**, it also includes **Date of Birth**.
- If you select **Driver's License**, it also includes all dependent driver licenses listed below.

__95. Start at the Navigation Menu.

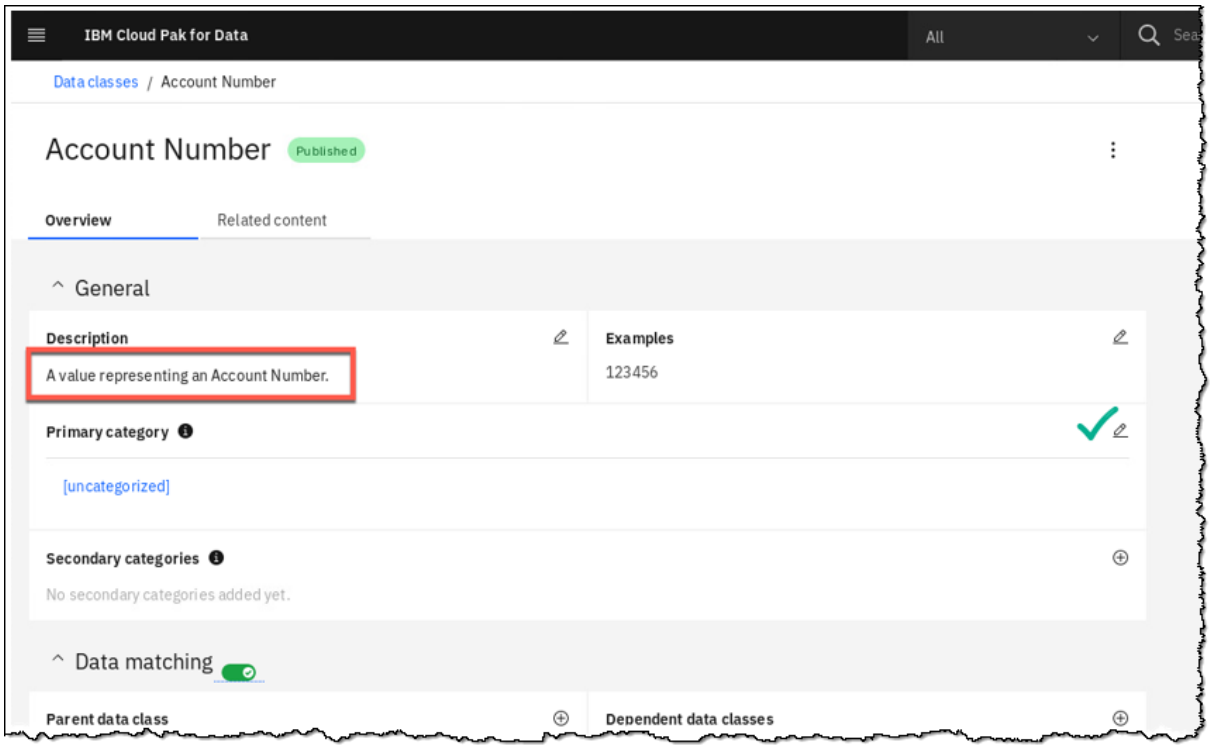
Click Organize ⇨ Data and AI governance ⇨ Data classes.



__96. Scroll to find Account Number and click on it.



__97. The Data Class is described here. You could also add the primary Category here.



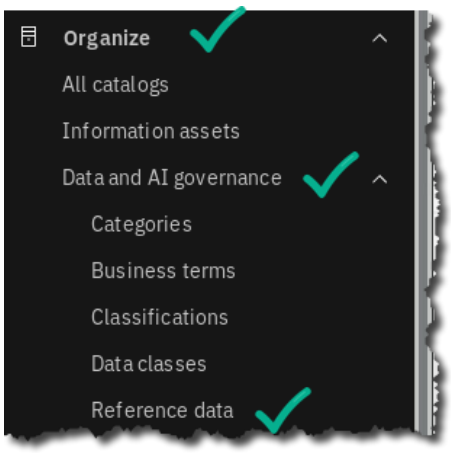
13.8.3 Reference Data Set

Reference Data Sets define lists of permissible values that are allowed for use within a data field and may be referenced by Business Terms, Policies, Rules and Data Classes in Watson Knowledge Catalog.

You can capture, manage, and socialize reference data — setting it up once and re-using the reference data in other places.

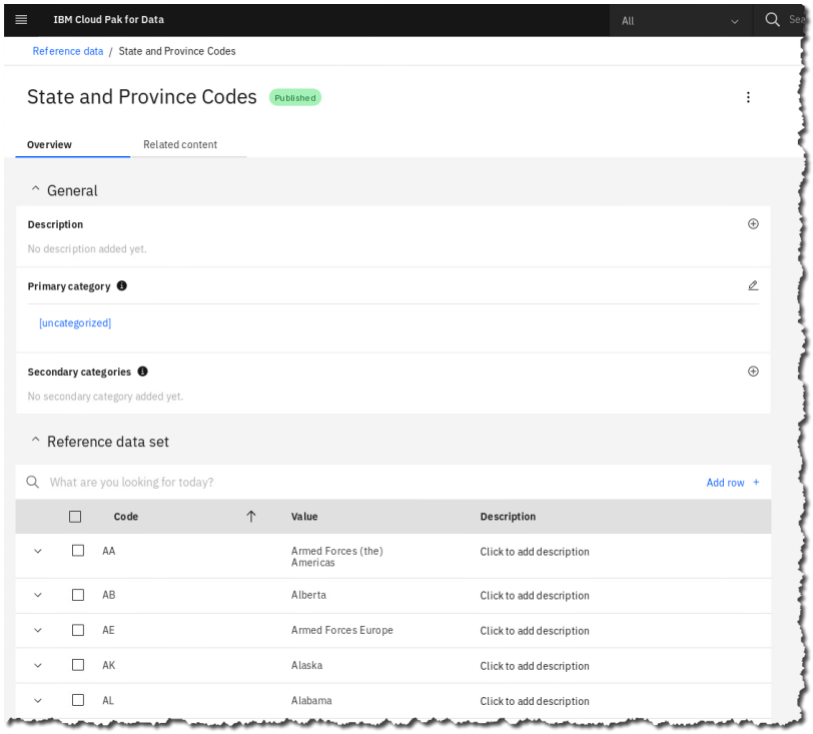
__98. Start at the Navigation Menu.

Click Organize ⇨ Data and AI governance ⇨ Reference data.

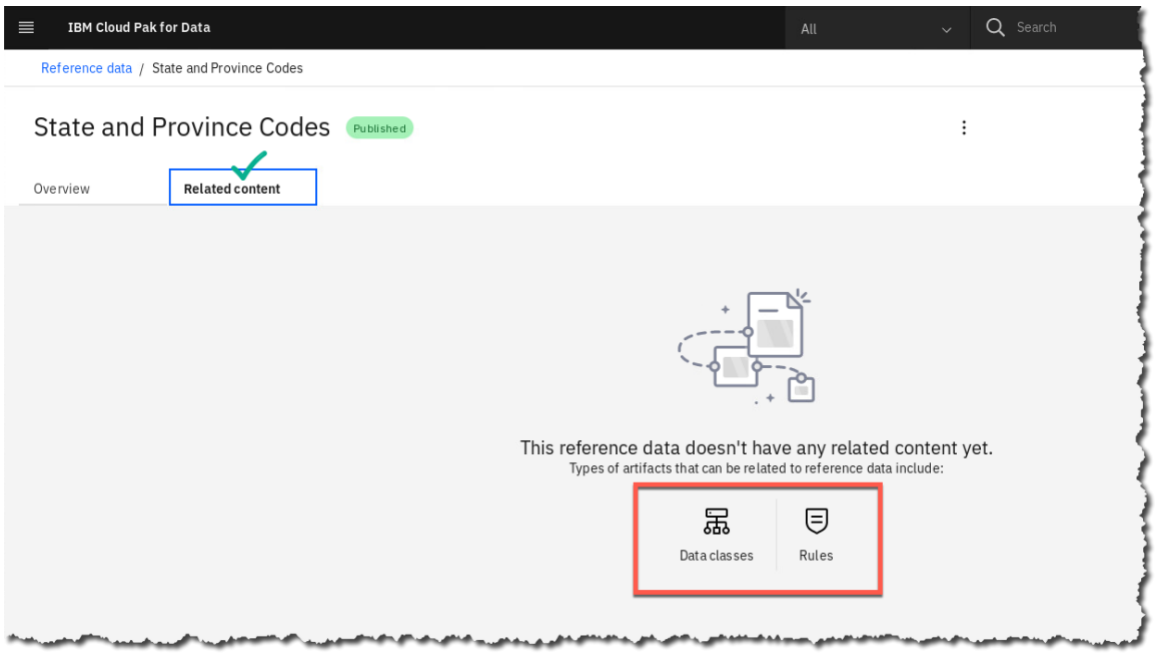


__99. Scroll to find Reference data set State and Province Codes and click on it.

__100. Scroll down to review the Reference data set rows (the data).



__101. Click section Related content and review how you can relate Rules or Data classes to it.



13.8.4 Data Protection & Masking

Another challenging requirement in Data Governance is the ability to ensure that only the correct users or roles see the correct data. Often, this can get very granular, such that a particular role needs to view a set of columns in a table, but other columns in the same table are off limits. Like with Business Glossaries, the challenge is in how to ensure data protection when there is an exploding volume of sources – enter automation.

Data Protection Rules are a unique and incredibly powerful capability in Cloud Pak for Data. They allow you to set up masking or redaction capabilities, on a columnar or tabular basis once, and any data that subsequently gets added to the catalog, if it has the same characteristics as the rule states, will automatically be masked or redacted. This is another benefit of the automated data discovery and classification process.

It works like this:

a. You build a data protection rule stating that any column in the data catalog that is classified as ‘Date of Birth’ must be redacted from view when the user is a developer.

b. You run data discovery on a new data source that you want to integrate into your catalog, data lake, etc.

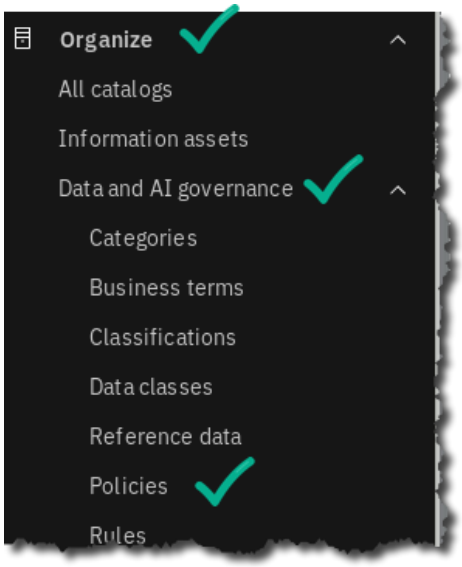
c. Data discovery automatically classifies any field it determines through ML and fuzzy logic to be DOB information with the data class “Date of Birth”.

d. You associate this data protection rule with a more general policy stating that certain data elements must be redacted.

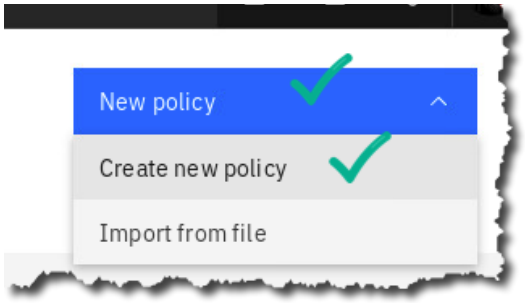
e. Immediately this data protection rule will be executed any time a developer tries to view a Date of Birth column.

Let’s see how this works in practice. The first thing we will do is create a policy around masking PII data.

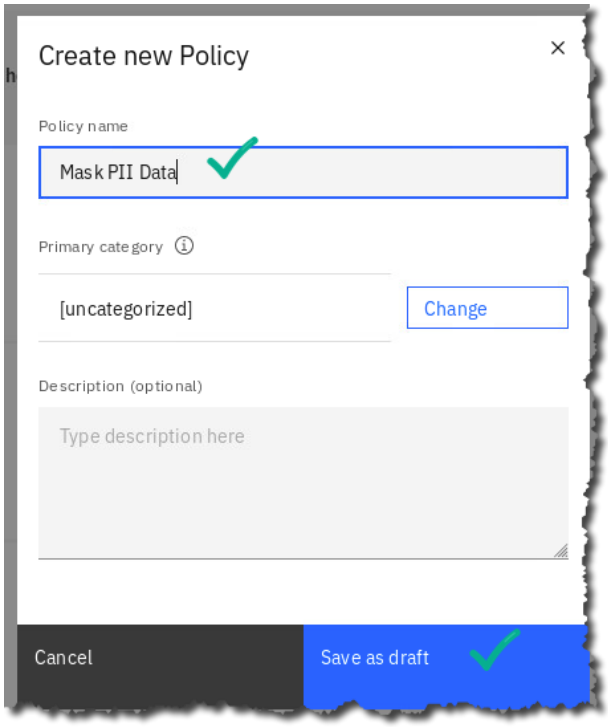
__102. Click Navigation menu ⇨ Organize ⇨ Data and AI Governance ⇨ Policies.



__103. We'll create a new policy, so click New policy ⇨ Create new policy.



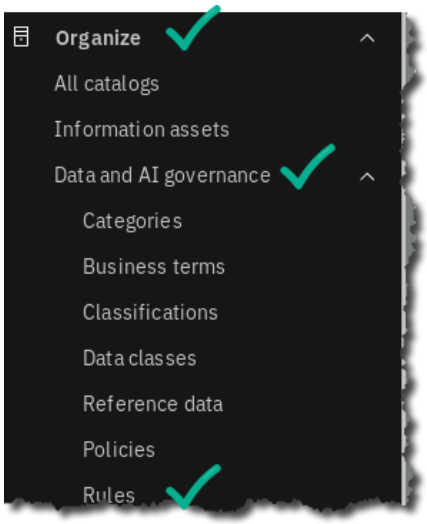
__104. Now fill in the following information (Policy name = Mask PII Data) and click Save as draft.



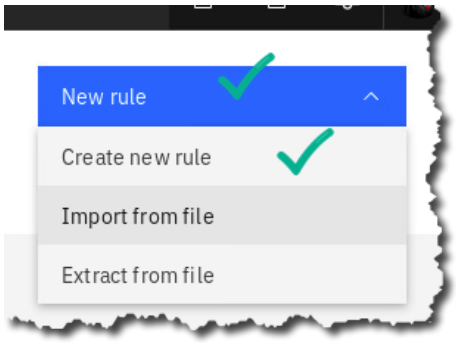
We save this as a draft because Cloud Pak For Data has a built-in workflow capability that ensures that all governance artifacts, rules, policies, etc. need to be reviewed before being published.

__105. Publish the policy – click Publish – Publish.

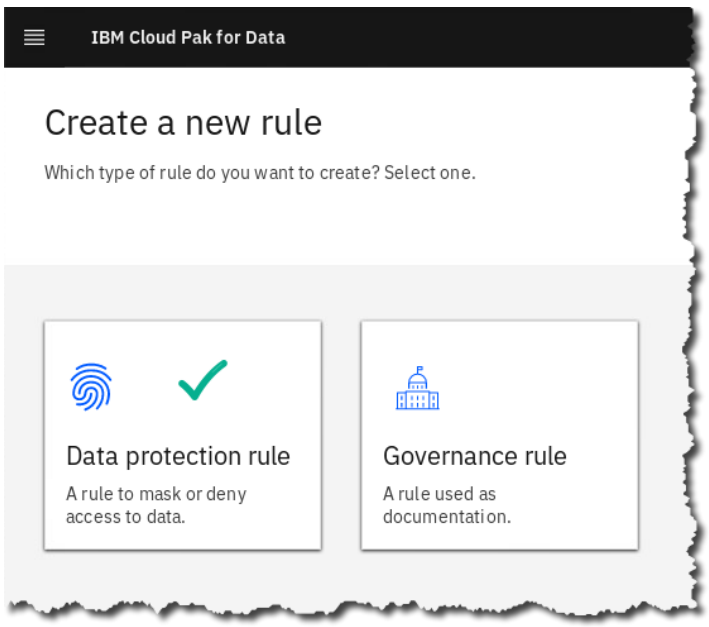
__106. Next, we'll create our data protection rule, and then link it to our policy.
Click Navigation menu ⇨ Organize ⇨ Data and AI Governance ⇨ Rules.



__107. Click Create new rule.



__108. Choose Data protection rule.



__109. Now we'll enter the criteria for our rule.

Fill out the form as shown here:

Name: Mask DOB for Developer user

Type: Access

Business definition: A rule to ensure that PII masks date of birth.

If Data class contains any 'Date of Birth'

and (+)

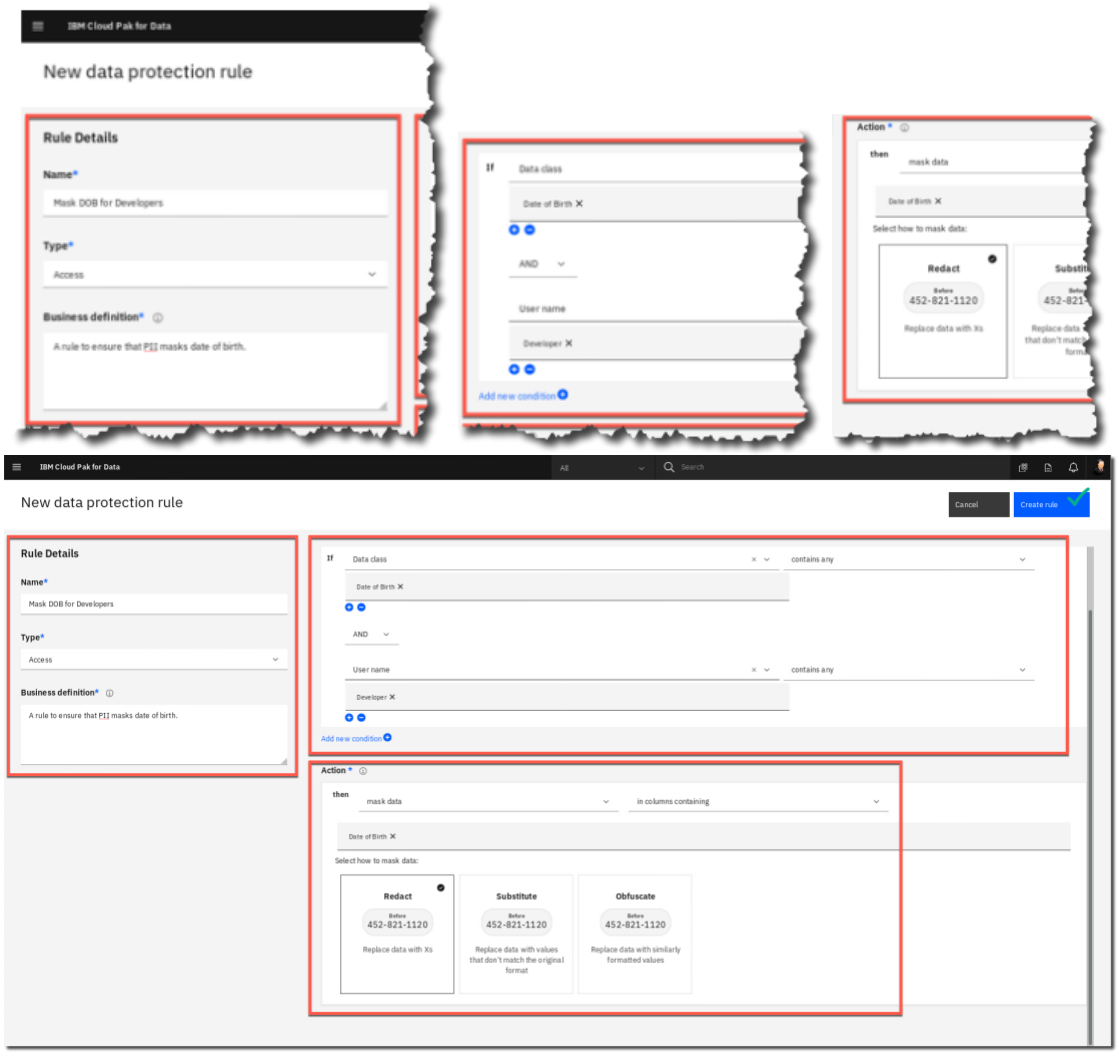
User name contains any Developer

Action

then mask data in columns containing Date of Birth

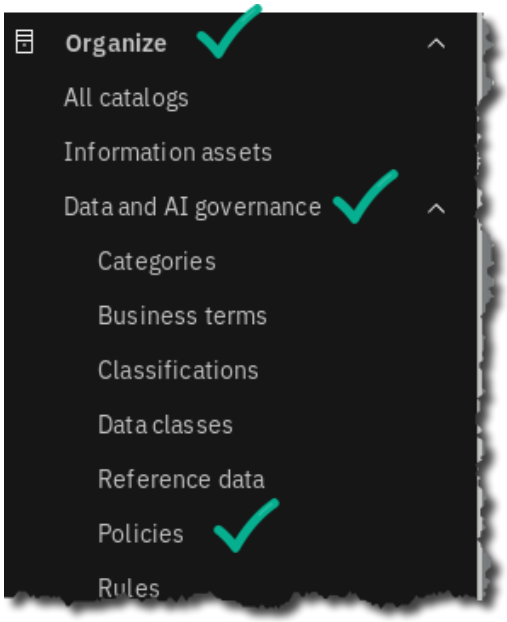
Redact

and click Create rule.

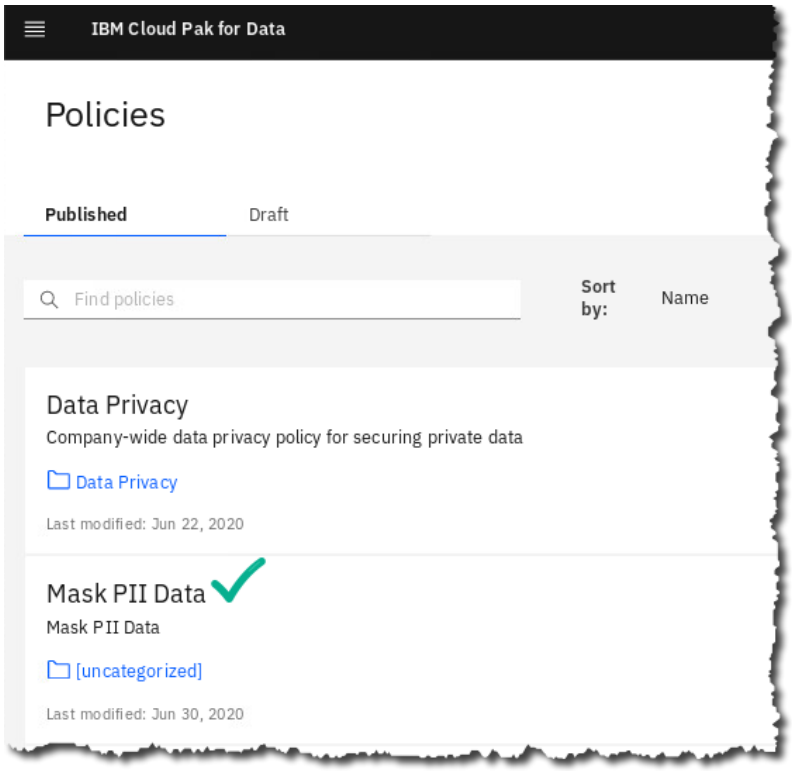


__110. Next, we want to associate this rule to the Policy we created earlier. For data protection rules to be enforced, they need to be associated with one or more policies; this ensures that there is a business-documented reason (the policy) for having the rule.

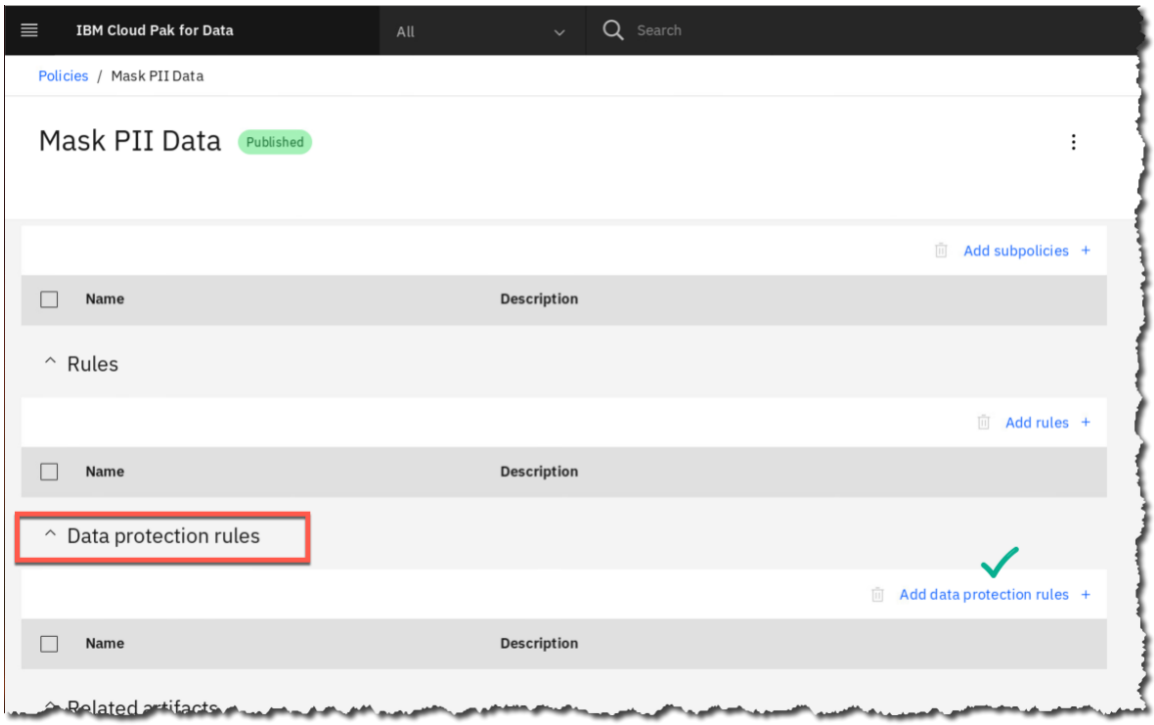
Click Navigation menu ⇨ Organize ⇨ Data and AI Governance ⇨ Policies.



__111. Click to open the policy called Mask PII Data.



__112. Next, scroll down until you see the section of the page called Data Protection Rules and click Add Data Protection Rule.



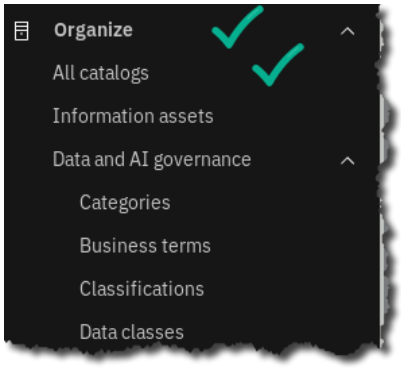
__113. Click the checkbox next to our Data Protection Rule.

Then click Add.

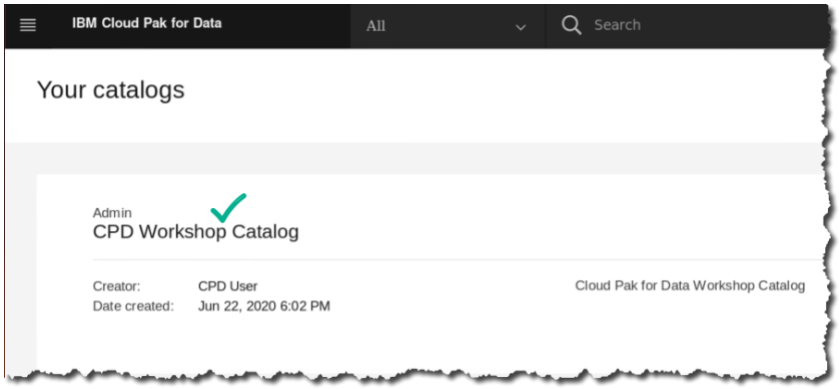
Note: There's no need to hit save after adding the Rule; it's automatically saved, and you will get a small indicator of that in the upper right of the screen.

__114. The next thing we'll want to do is add a collaborator to our catalog – catalogs can have collaborators with various roles; we will add our 'developer' user, and then we'll log in as that user to see the masking in action.

Go to Navigation Menu ⇨ Organize ⇨ All catalogs.

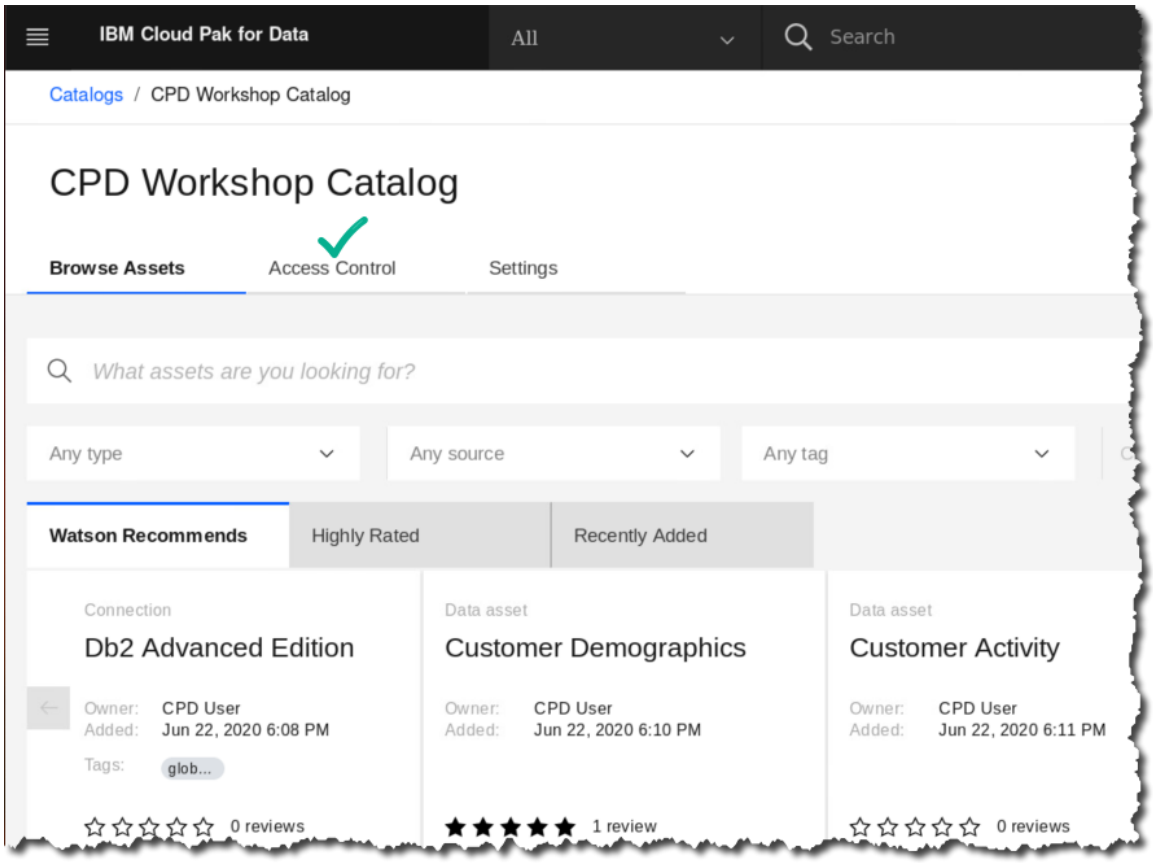


__115. Click on CPD Workshop Catalog.

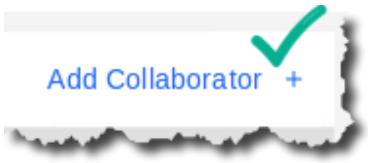


 Data Steward	Note – This is the catalog where our data assets were automatically published after running Auto Discovery.

__116. This will bring you into the screen that shows the assets in the catalog, and tabs for different types of information. Click the tab Access Control.

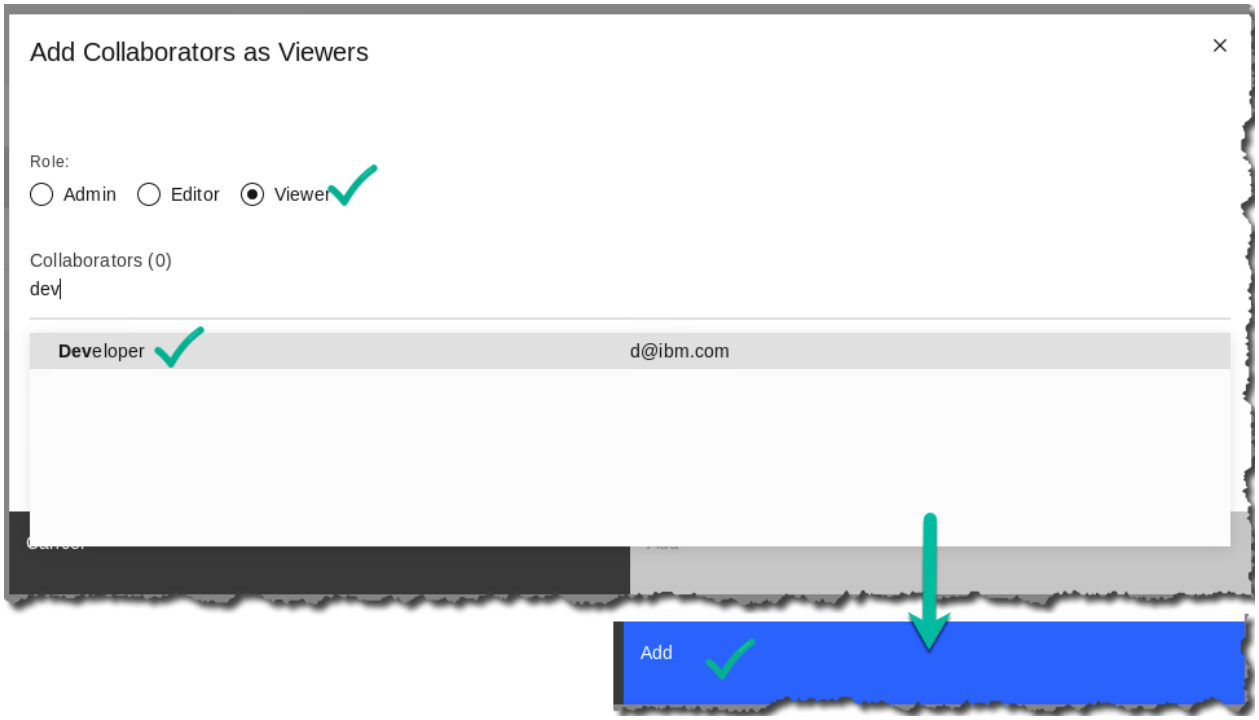


__117. Next, click the link on the right side of the screen that says Add Collaborator.



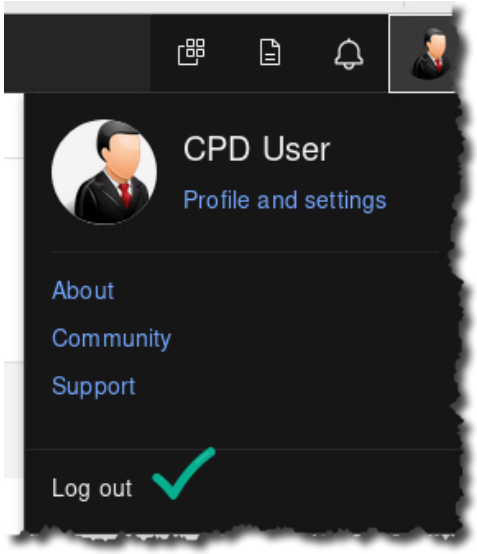
__118. Next, start typing the word Developer and you will get a drop down of all the users in the system with that in their name.

Choose Developer from the list ⇨ click Add. This will add the Developer as a Viewer (default).



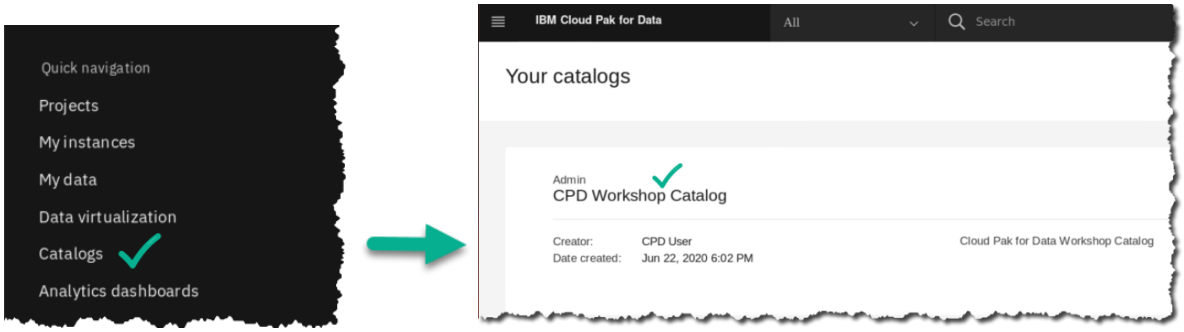
__119. We've now added the user as a collaborator to this project. Let's log in as that user.

First log out CPD User by clicking the User Icon in the upper right corner ⇨ Log out.

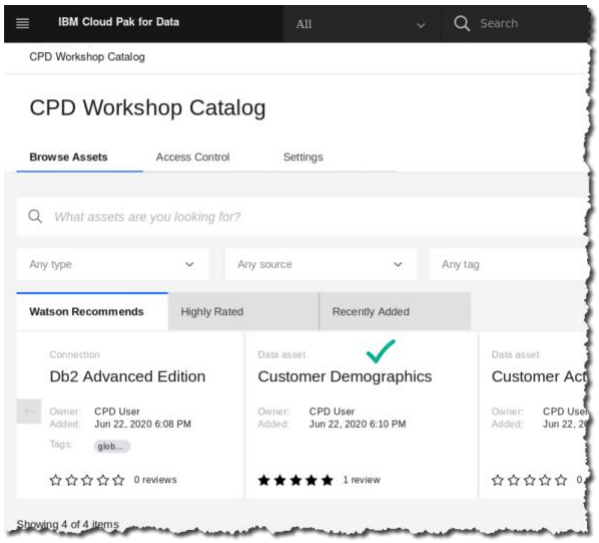


__120. Then log back in as developer (password should be saved in the browser, but it is cpdaccess).

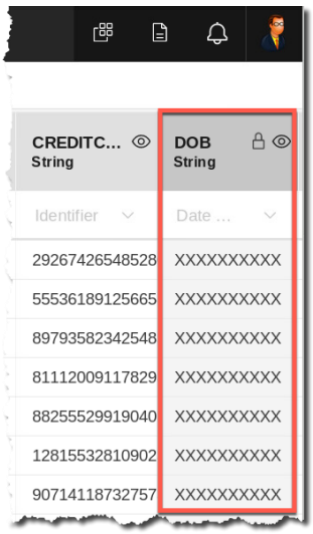
__121. When the welcome window comes up, click Explore Catalogs, and on the next screen open the catalog CPD Workshop Catalog.



__122. Next, we'll choose the table CPDUSER.CUSTOMER_DEMOGRAPHICS.



__123. When the Table preview opens, you will see that the DOB field is all 'X's' and there is a padlock icon over the column.



.

__124. Now log out and then log back in as user cpduser with password of cpdaccess.

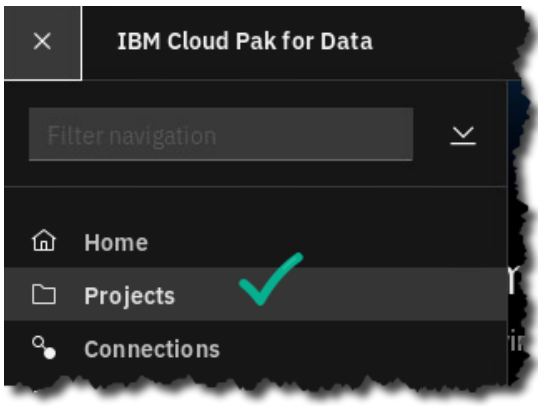
13.9 Transforming Data

One of the basic capabilities necessary in a good information architecture is the ability to implement sophisticated transformation and data manipulation. This capability is traditionally known as Extract/Transform/Load, or ETL. In Cloud Pak for Data, because of the strong governance capabilities, the ETL process automatically creates a documented data flow that enables data lineage. Data lineage is the ability to see exactly what happens to data as it moves through your organization's systems. This information is critical in today's world of data privacy regulation, where the ability to document your organization's data processing steps is a requirement for compliance.

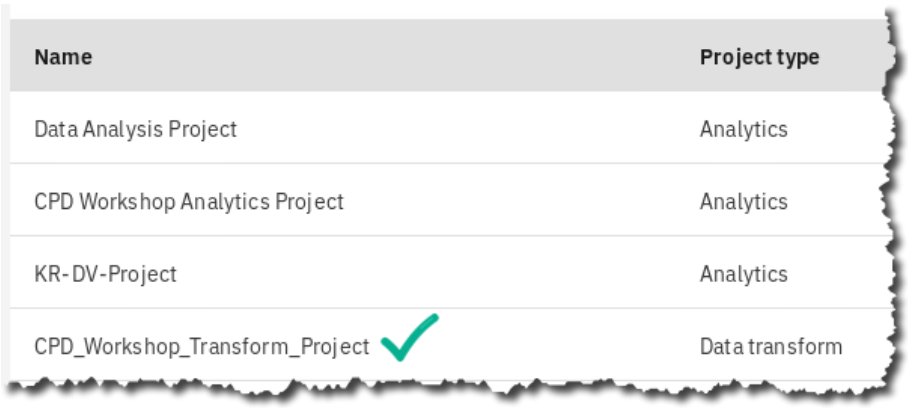
13.9.1 Reviewing an existing job

__125. ETL processes, or 'jobs', are developed in a special type of project called a Transform Project. So, let's open our transform project and get started.

Click Navigation Menu ⇨ Projects.

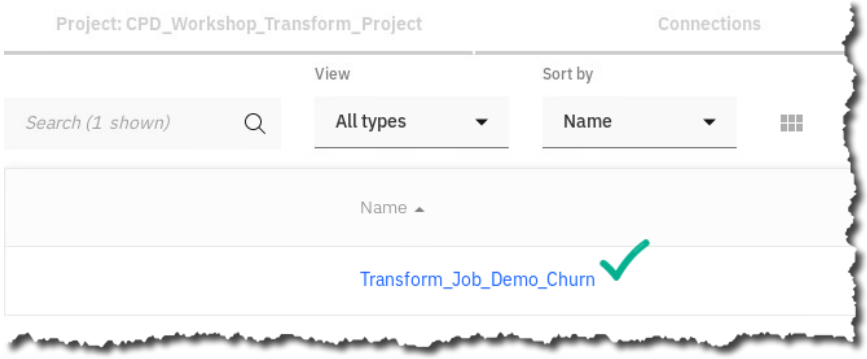


__126. Open the project CPD_Workshop_Transform_Project.

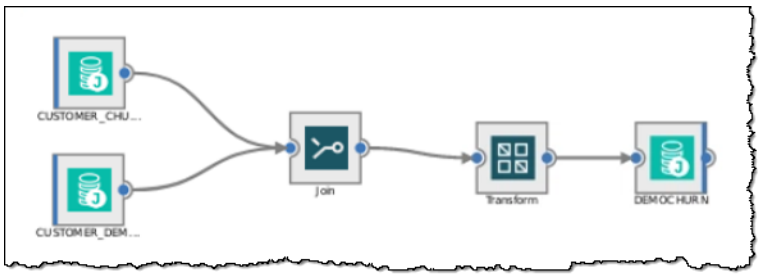


__127. When you open the project, you are shown a list of any jobs that exist. There are also tabs along the top of the screen for the different types of artifacts that are typically used in developing jobs – Connections, Table Definitions, Parameter Sets, and Jobs

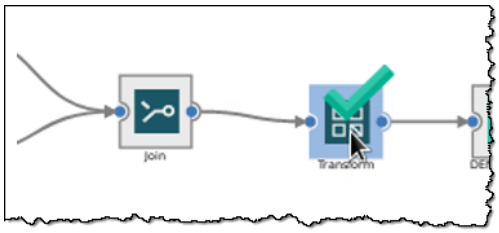
Let's inspect the job that's there. Click on the name to open it:



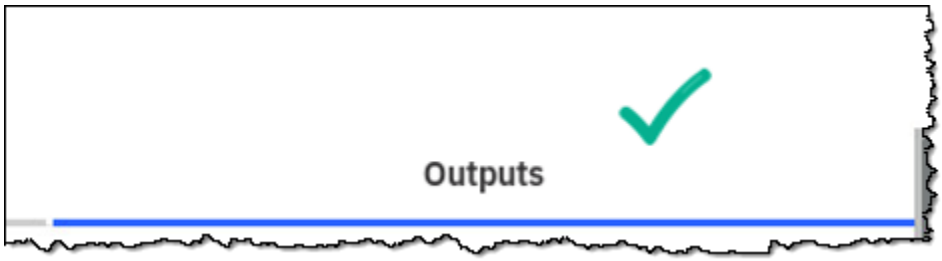
__128. When we open the job, we see a graphical data flow moving left to right. This job actually joins two tables, then does some work to transform some of the columns, and finally writes out the joined data to a new table.



__129. Double click on the Transform stage – this opens up a window that shows our transformation logic.



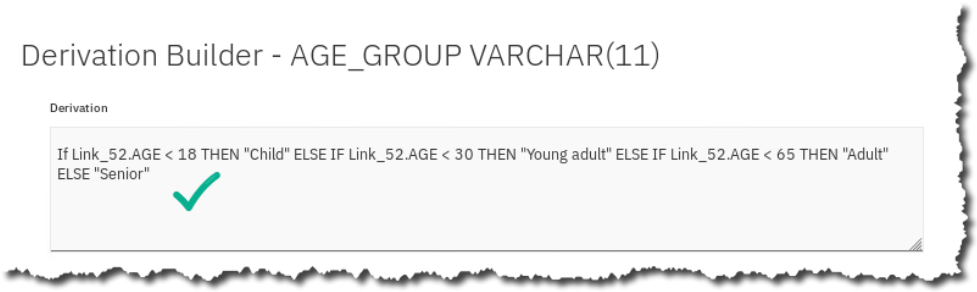
__130. Click on Outputs.



__131. Now we see a grid with Columns for derivation, column name, etc.

Scroll all the way to the bottom and we see a row with a derivation, for the column AGE_GROUP.

__132. Double click to see the full derivation; in this case we're creating a column to hold a value that describes the age group of each individual.



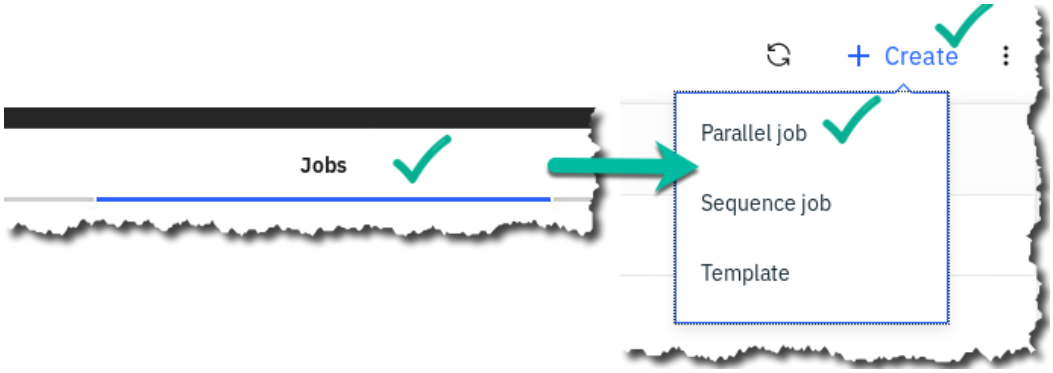
__133. Cancel out of the stages until you're back to the job flow diagram.

Then click the x in the upper right corner to close the job.

13.9.2 Creating a job

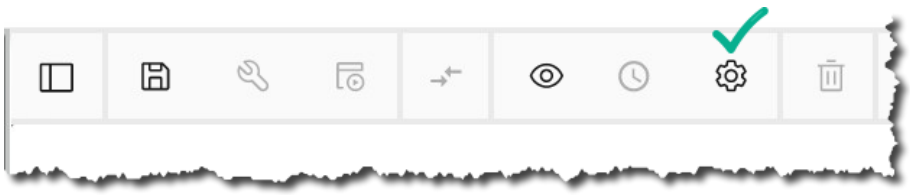
We will next create our own Transformation job.

__134. Click Jobs Tab, then Create ⇨ Parallel job.

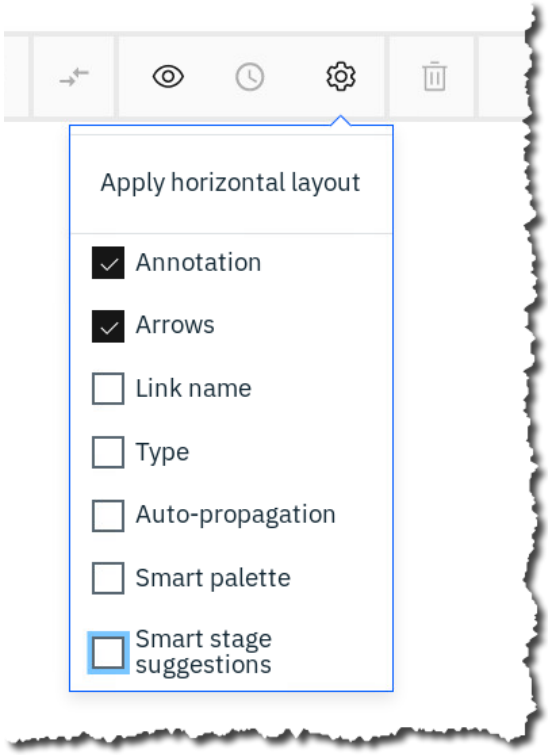


__135. This created a blank canvas we can use to develop our job. Before we begin, first ensure the settings are correct.

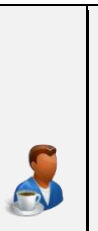
Click the gear icon for Settings.



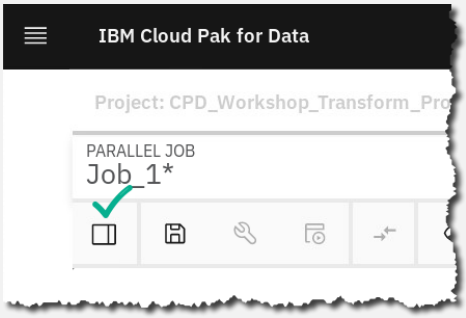
__136. Then ensure the list shows the same checked and unchecked options as below:



__137. On the left you see all of the different stages available for connecting to data, transforming it, routing it, etc.



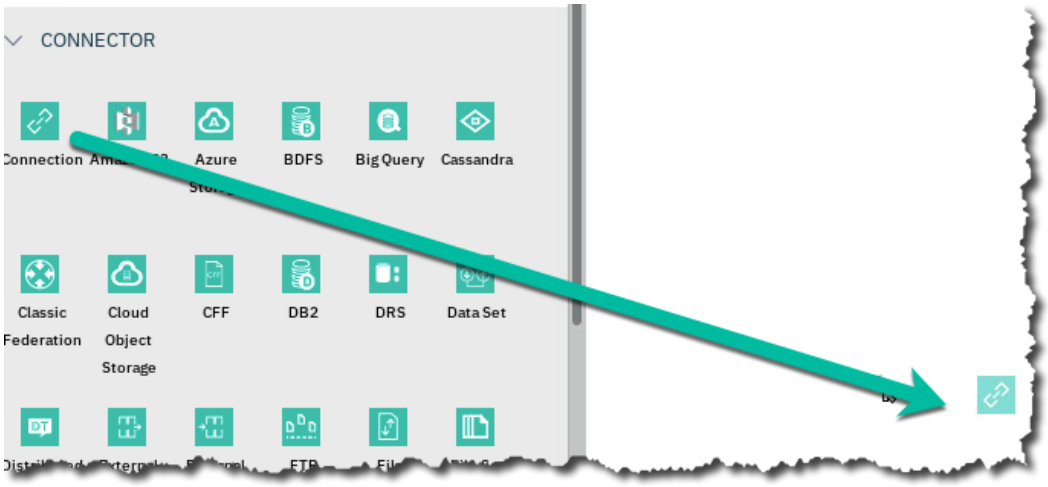
Note: If the palette isn't showing, click the leftmost icon on the button bar:



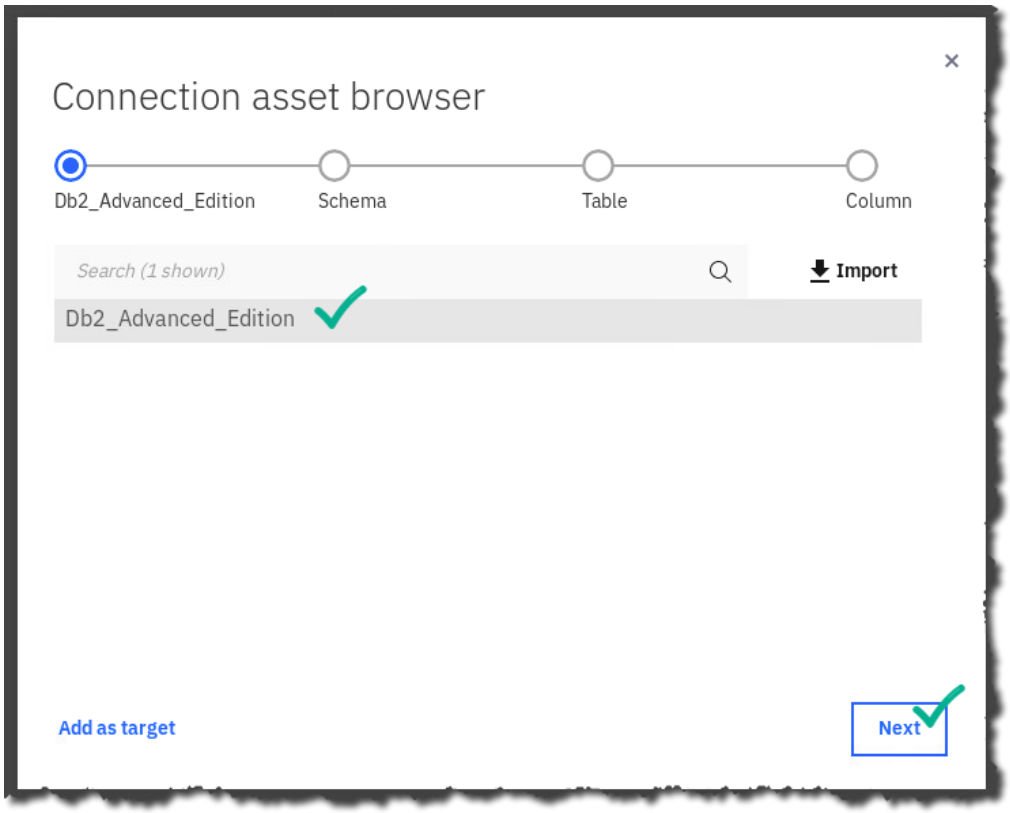
__138. We're going to create a job that joins our two tables, CUSTOMER_DEMOGRAPHICS and CUSTOMER_ACTIVITY. Let's start by dropping two stages to represent these tables.

On the palette, find the Connection icon. It should be the first one on the palette.

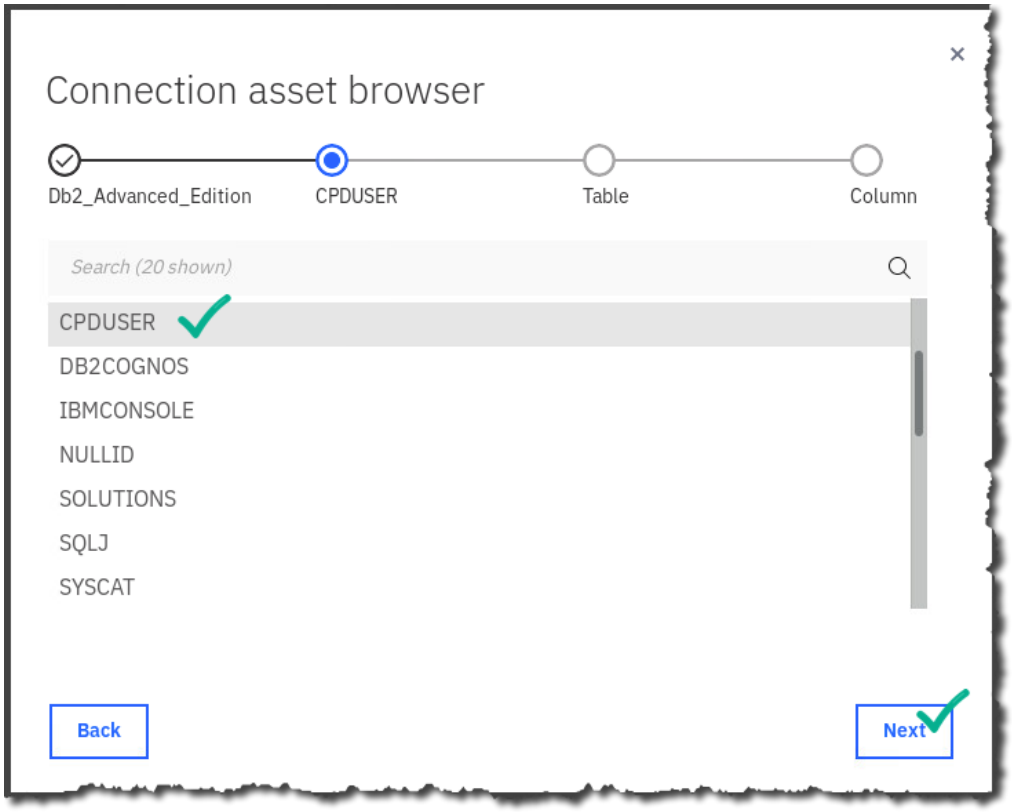
Left click the icon, drag it to the canvas, and left click again to release it. This will open up a window where you'll be guided to fill out the connection information for the database we'll be extracting from.



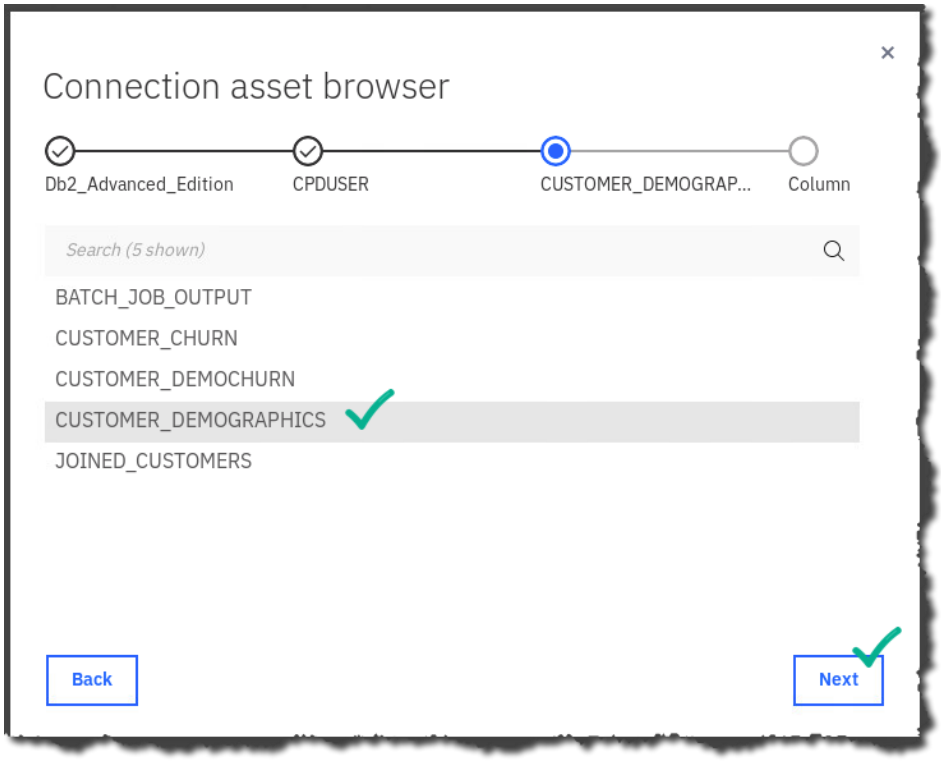
__139. The Connection asset browser appears. Click Db2 Advanced Edition to select our Db2 Advanced Edition connection, choose it from the list and then click Next.



__140. On the next window, choose Schema CPDUSER, and click Next.

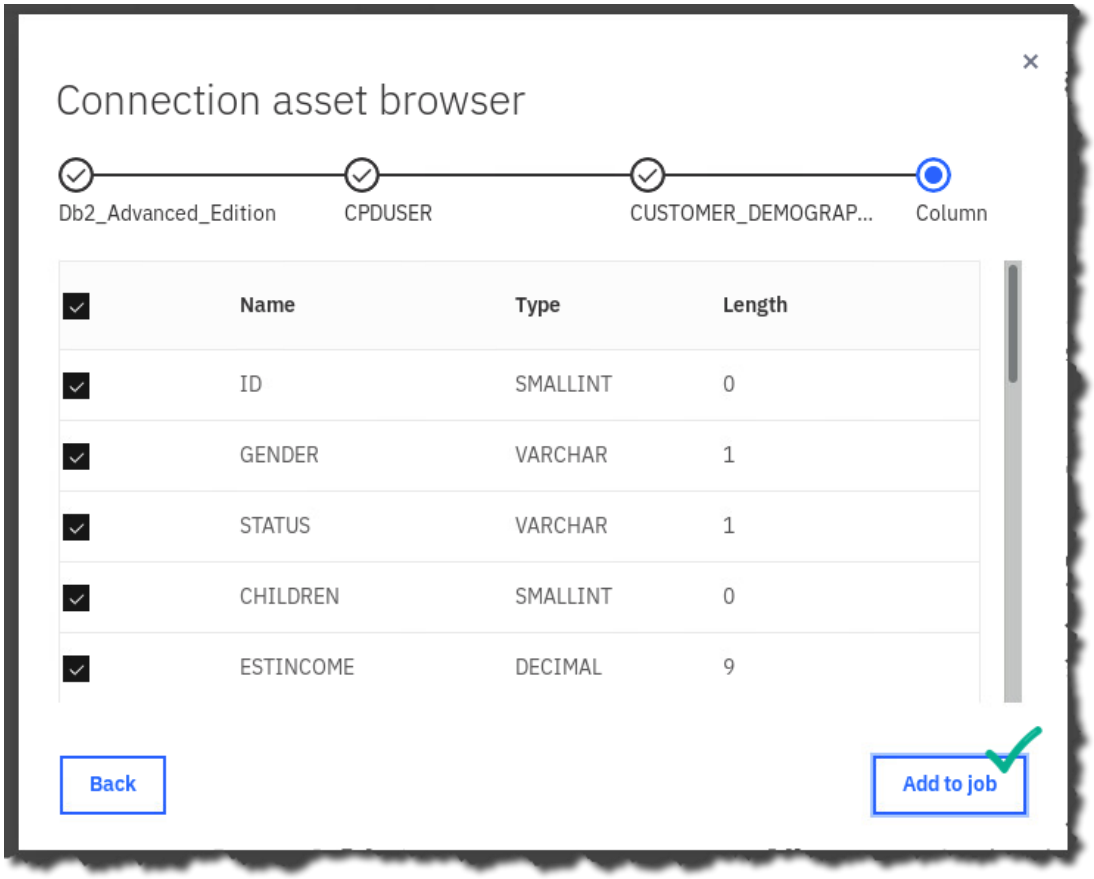


__141. Next, it's time to choose our table – we're going to choose CUSTOMER_DEMOGRAPHICS and then click Next.

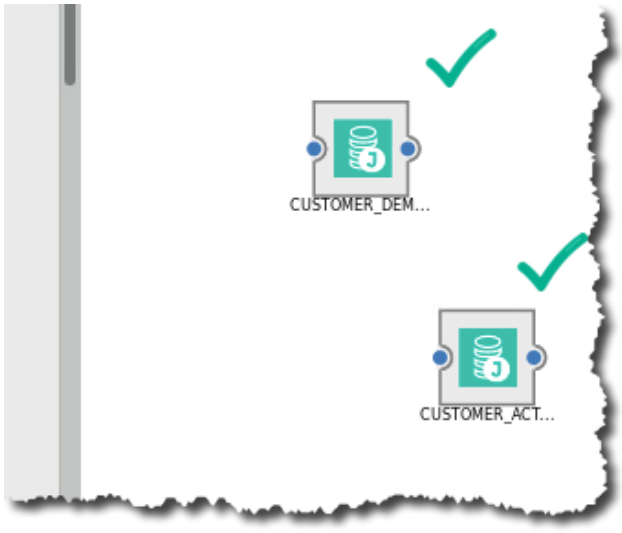


__142. Now we see a list of the columns in our table.

Click Add to job.



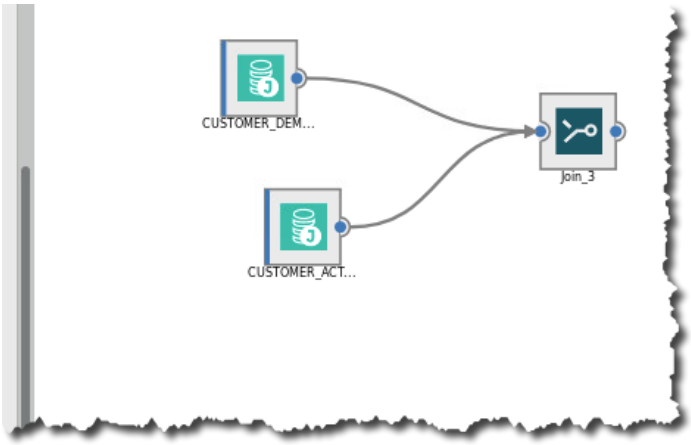
__143. Now, we'll do the same process for our second table. Repeat the same steps you just did, but this time, change the schema you choose to 'SOLUTIONS' and the table name to 'CUSTOMER_ACTIVITY'. When finished, your canvas should look like this:



__144. Single click the CUSTOMER_DEMOGRAPHICS icon on the canvas.

__145. Navigate to the palette on the left and do the following:

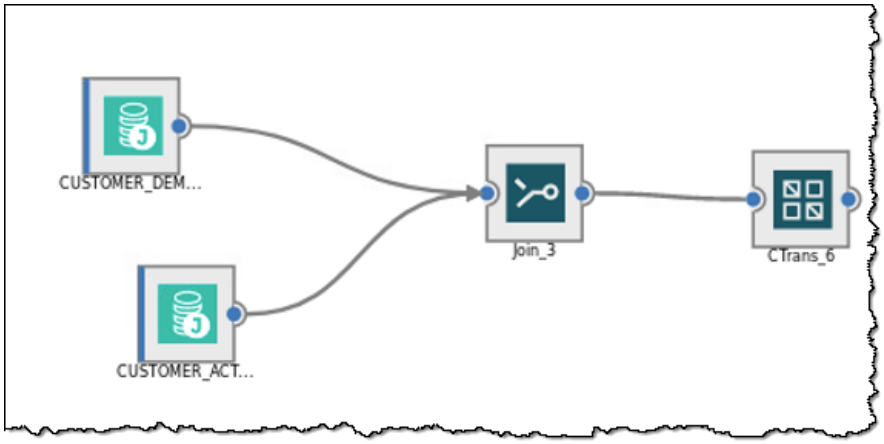
- Find the Join icon on the palette (in the Stages section.).
- Click it once.
- Hold your mouse button down and drag the stage to the canvas.
- This should draw a link from CUSTOMER_DEMOGRAPHICS to the join stage. [If you do not see the link being drawn, remove the Join and make sure that you click CUSTOMER_DEMOGRAPHIC first.]
- Next, single click on the CUSTOMER_ACTIVITY stage.
- Click the small blue button on the stage, then drag a link to the join stage.
- You should see the join between CUSTOMER_DEMOGRAPHIC and CUSTOMER_ACTIVITY.



__146. If you do not get it right it the first time, delete the Join from canvas and try again.

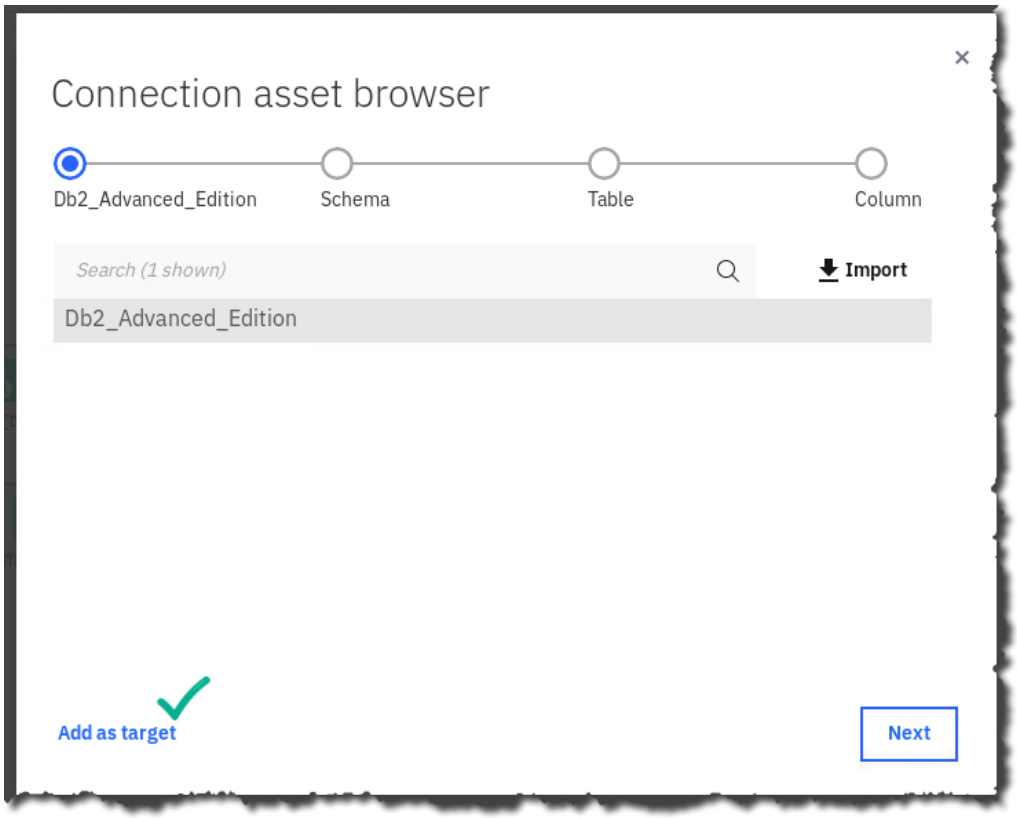
__147. Single click Join on the canvas to select it. (Make sure you select it before going to next step.)

__148. Next, go to the palette, select the transformer stage, and drag it onto the screen.

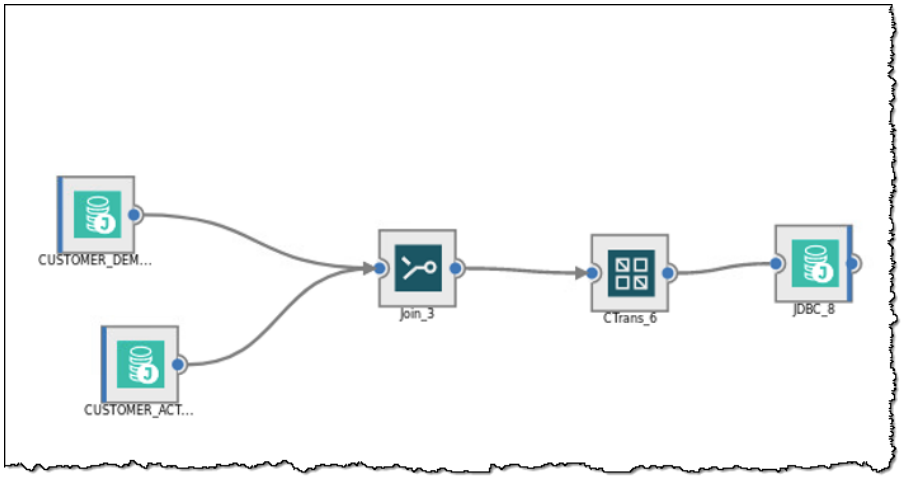


__149. Lastly, we'll add our target stage. Single click on the Transform stage. Then select the Connector icon at the top of the palette, and drag it onto the canvas. This will pop up the connections configuration window.

__150. Click the link at the bottom that says Add as Target. This should immediately link the target stage with the Transformer stage, but if not, draw the link from the transformer stage to the target.



You should see the following:



__151. Next, double click the last connection icon.

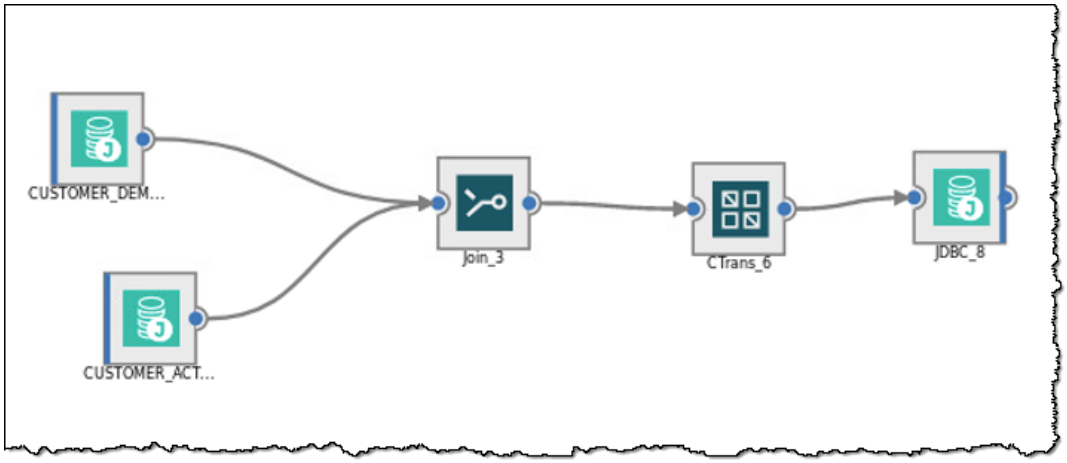
This will bring up the property sheet for our target table. We'll want to put in a table name, and we'll also want to create the table as part of the job run if it is not there.

__152. First, under the Table name field, enter CPDUSER.JOINED_CUSTOMERS, and then under Table action choose Replace.

This will attempt to drop and re-create the table, but if no table of the same name exists, it will just create the table.

Click OK to save the properties.

__153. Your job should now look like this:

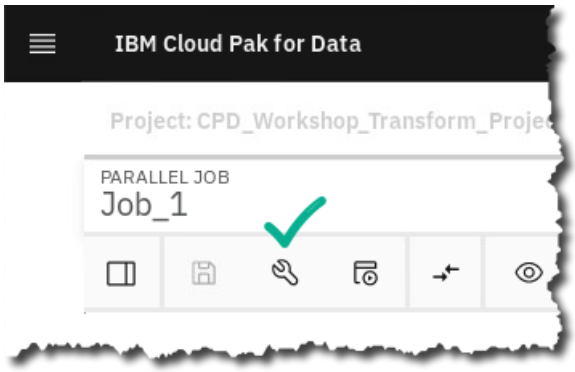


__154. Next, we'll save, compile and then run the job.

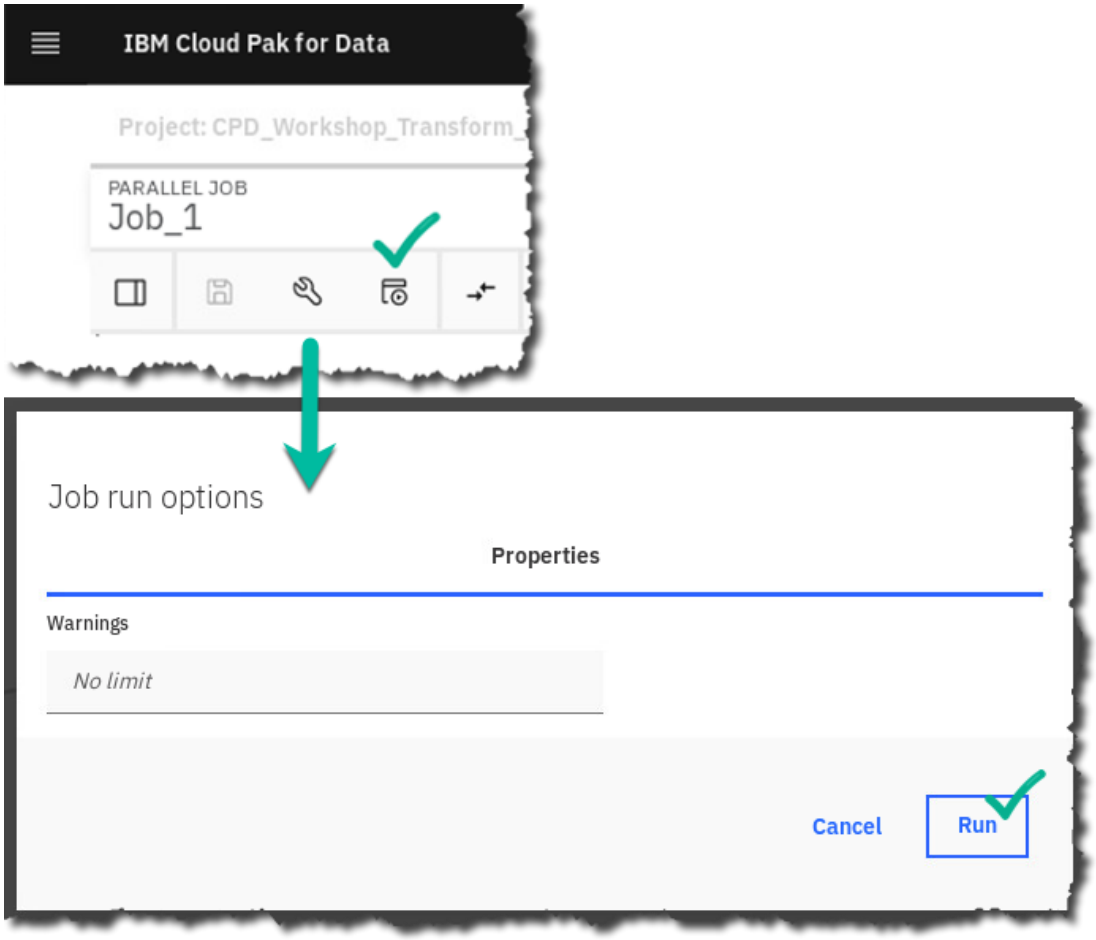
First, click the diskette icon to save the job.

Give it a name (or take the default Job_1) and click Save.

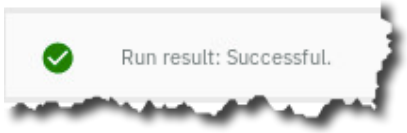
__155. Next, we'll compile the job by clicking the wrench icon.



__156. Begin to run the job by clicking the run icon, then Run the job.



__*157.* When the job is done you'll see a green check mark and a run result.
(Note: A Warning message may appear that may be safely ignored.)



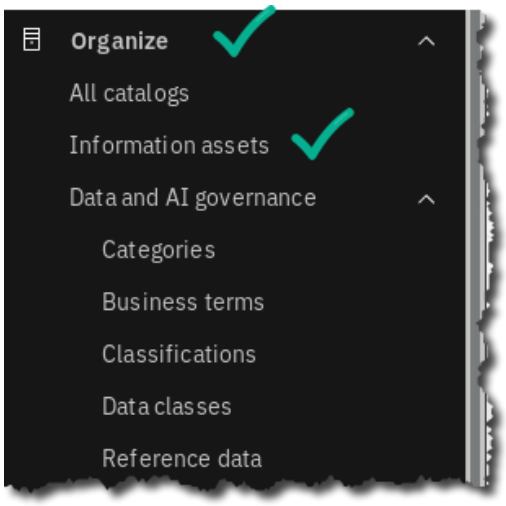
Lastly, if you'd like to see the result, double-click on the target stage (the rightmost stage) and then click View link in the property sheet. This will show us our joined result.

13.10 Data Lineage

The last thing we will explore is Data Lineage. One of the main benefits of a well governed Information Architecture is the ability to describe and document how data flows through your organization, and what transformations are executed on the data. This allows organizations to better comply with regulations, understand the impact of changing different components of their Information Architecture, and a host of other benefits.

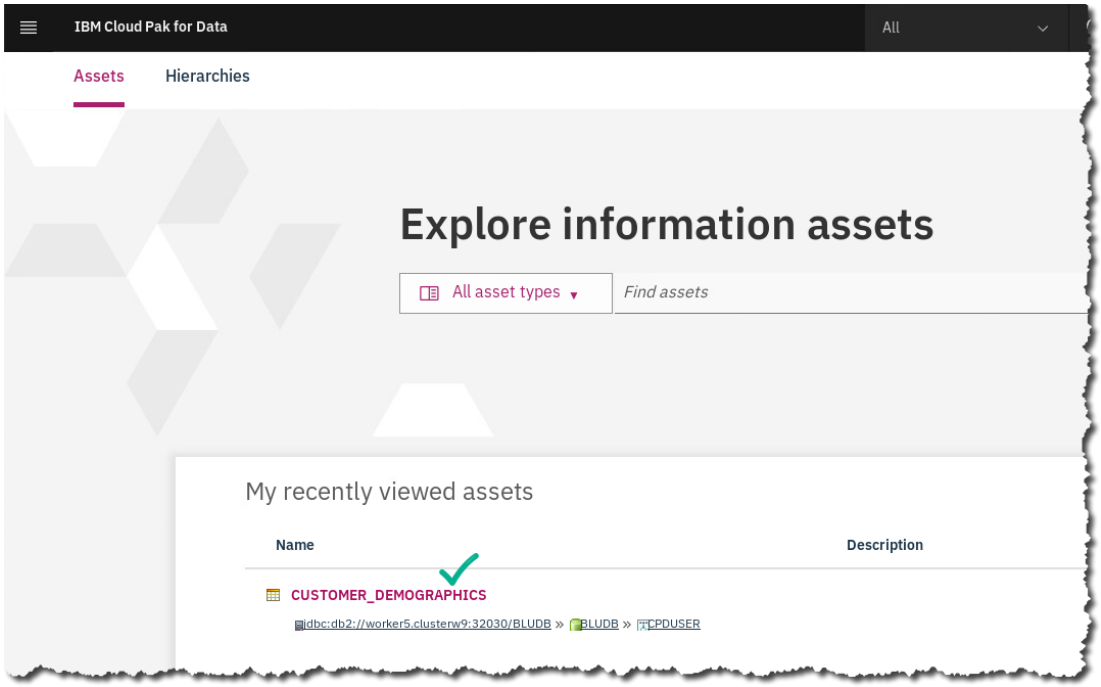
__158. The job we have just built and run automatically documents our data flow and allows us to easily visualize what happened during this job run. Let's go see what this looks like:

Navigate to Information Assets.



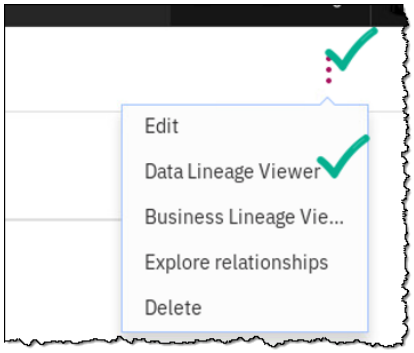
__159. This will bring up a screen that shows you recently accessed assets, and a search bar.

Click on the table CUSTOMER_DEMOGRAPHICS. If the table is not there, search for it by typing DEMO into the search bar.



__160. Once chosen, a window showing table details is displayed.

Click the ellipses (3 vertical dots) in the upper right corner, and then click Data Lineage Viewer.



__161. Click Run Lineage.

__162. You will then be brought to a visualization of the flow of the transform job we created.

Notice that our source table, CUSTOMER_DEMOGRAPHICS, is actually a source for two separate jobs, and that those jobs write the data to two separate target tables.

You can further explore and drill into these steps to see more detail. This is an incredibly powerful capability, which is automatically available by virtue of the Transform design process.

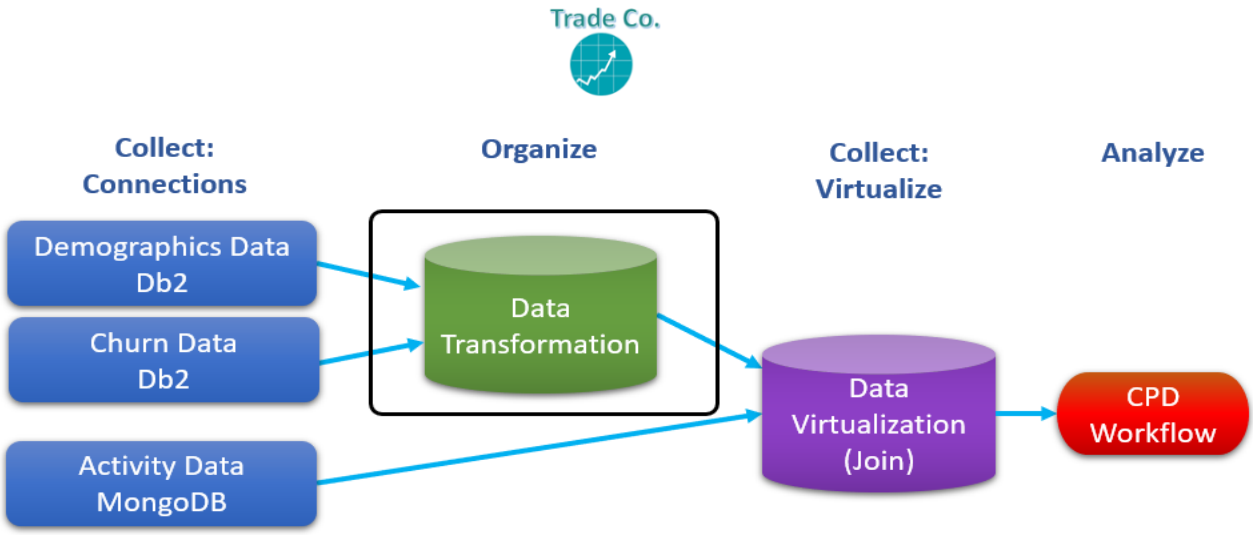
13.11 Lab conclusion

We have seen the value in creating a Data Dictionary by creating a glossary of categories and terms to make data searchable so that data scientists, data engineers and business analysts can shop for data. The CPD platform reduces the time-consuming data organization task, making it easier for the consumers of the data to access and analyze the trusted data they need.

This lab has shown you how to automatically discover data sources, automatically classify those sources with business classifications using CPD machine learning methods, and automatically assign business terms to those data sources.

This lab also showed you how to create data protection rules, and join and move data to an analytics Advanced Edition in a Data Transformation step for further use by the data consumers in the CPD workflow.

The steps covered here could normally take many weeks, months, and sometimes even years, to complete using traditional manual methods. Cloud Pak for Data automates these operations so that you can accelerate the time to value of your organization's analytics projects.



**** End of Lab 13 – Organize – Deeper Dive**

Lab by John Van Buren, Burt Vialpando and Kent Rubin

13.12 Additional Optional Activities

There are many other capabilities within the 'Organize' section of Cloud Pak for Data. Here we will highlight one of these.

Using Data Virtualization sources in Organize

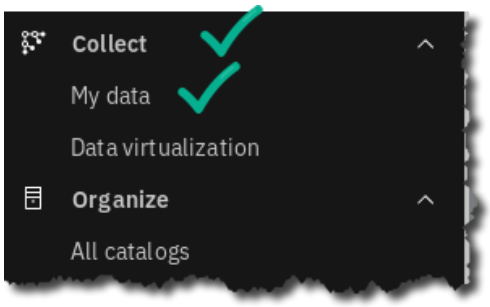
One of the strengths of an integrated platform is that you can use the various capabilities interchangeably, reuse previously designed components, and in general create a data fabric more quickly than with individual tools.

In an earlier lab, we created a virtualized view that was a join of a Db2 warehouse table and a MongoDB document store. What if we wanted to run auto discovery on that view to understand the quality of the data in those two sources when joined together? Or what if we wanted to automatically document the attributes of the view with Business Terms from our Business Glossary? Let's go ahead and do it!

First, we need to make sure we've got our virtualized view created.

NOTE: *This view was created during the execution of Lab 3, document:* ***LC03-CPD-3.0.1-Collect-Connections.docx***

__163. You can verify it is there by doing the following:
From the main navigation menu, choose Collect→My data.



__164. This will bring up a list of virtualized tables and views. Look for the one with the name USER1001.CUSTOMER_DEMOCHURN**;** this is a view that represents the join of our DEMOGRAPHICS table, in Db2 with our ACTIVITY table, in MongoDB:

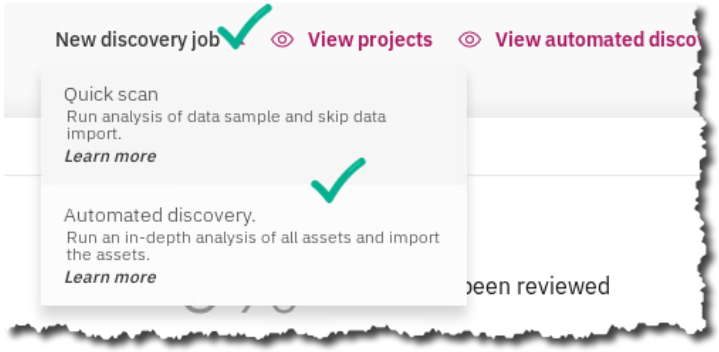
__165. If that asset is there, you can proceed to the next step.

Data Steward	Note: If it's not there, please go back and run the Collect and Virtualize lab from our CORE.

__166. Once you've established that your virtualized view is there, we want to simply run some of the actions against it the way we would against any other table.

__167. From the main navigation menu, choose Organize ⇨ Curation ⇨ Data discovery.

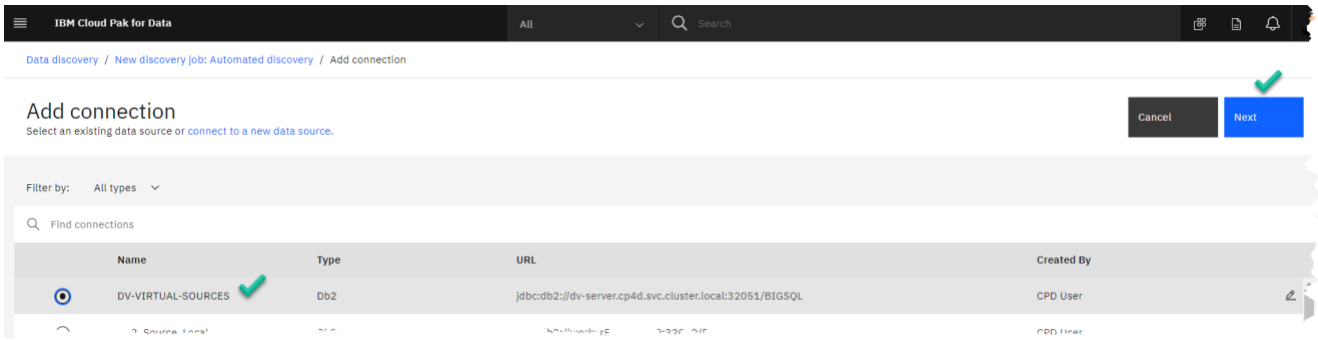
__168. Here, we'll want to run an automated discovery just like before – choose new discovery job→automated discovery



__169. Next click the 'Select a Connection' dropdown, and choose 'Add a connection'

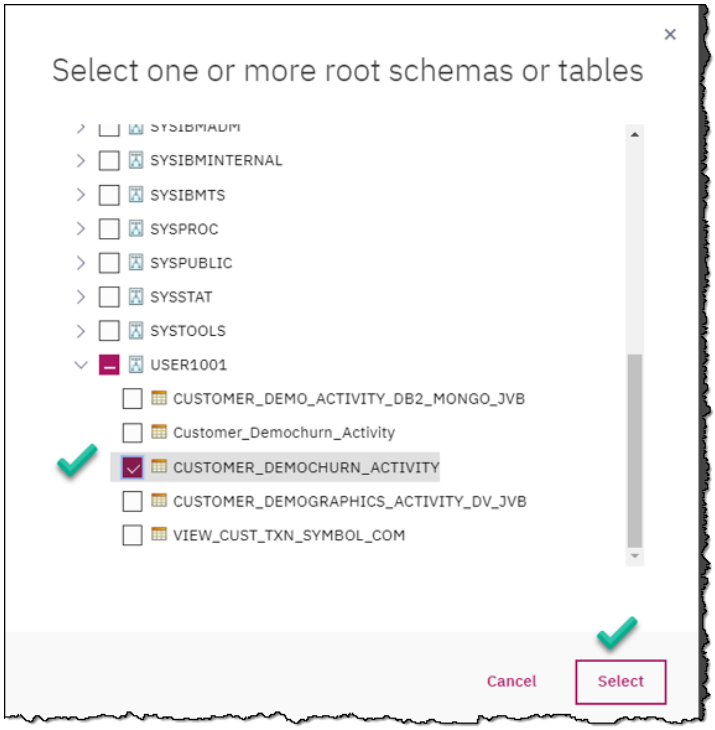
Data Steward	Note: if the connection 'DV-VIRTUAL-SOURCES' appears in the list, then you can skip adding a connection. Simply choose 'DV-VIRTUAL-SOURCES' from the list.

__170. The screen will change to a list of possible connections, choose 'DV-VIRTUAL-SOURCES' and then click 'Next':

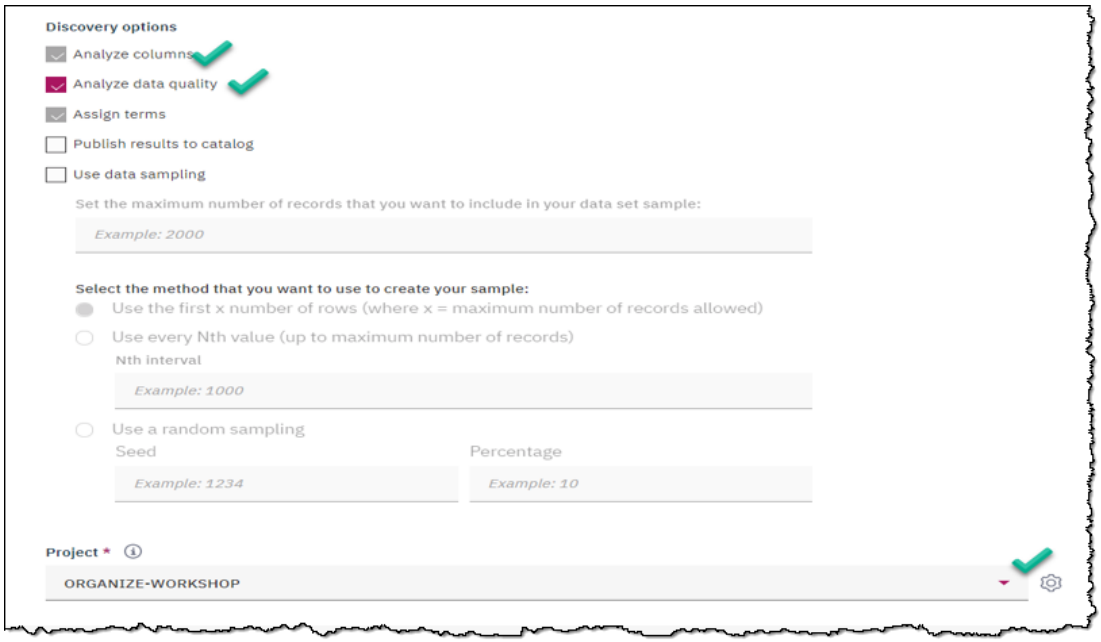


__171. This will send you back to the discovery window – click the Browse button to take a look at the data available –

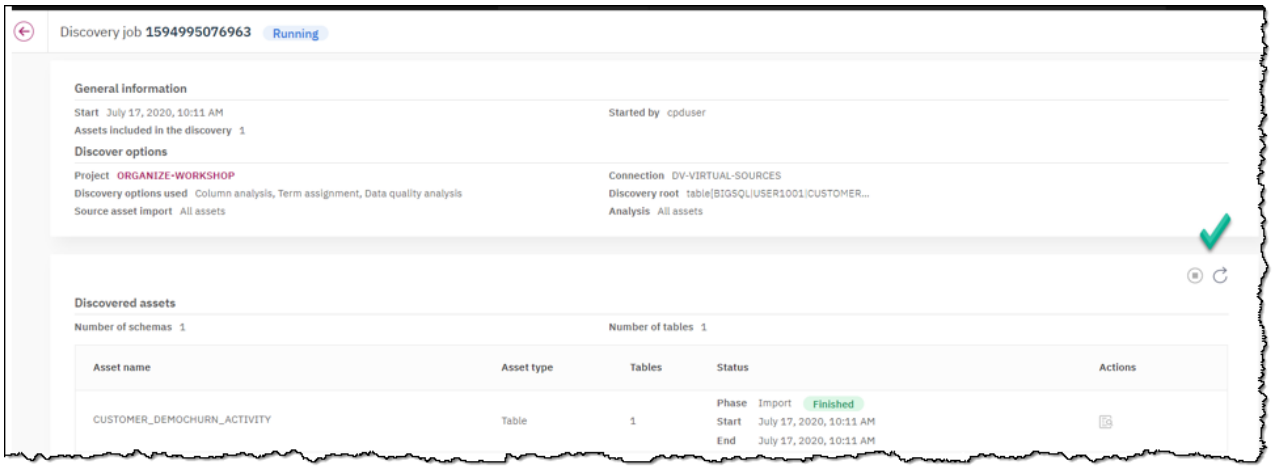
__172. This will bring up a tree view of the database (BigSQL) and the member schemas. Navigate down until you come to 'USER1001' and choose our virtual view, CUSTOMER_DEMOCHURN_ACTIVITY, and click Select



__173. Next, click the checkboxes for ‘Analyze columns’ and ‘Analyze data quality’, and then drop down the list of projects and choose ‘ORGANIZE-WORKSHOP’:



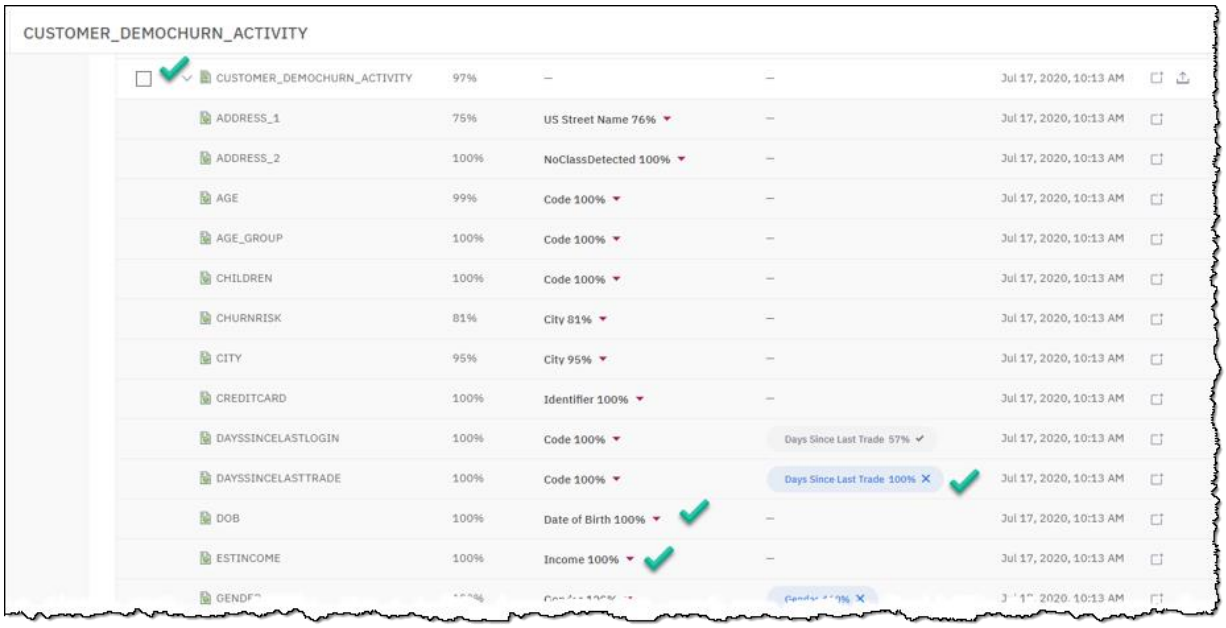
__174. Click the ‘Discover’ button. This will start the discovery process and you will see a status window you can click the refresh icon periodically to see when it is finished. :



__175. It will first show a status of phase Analyze as ‘running’:

__176. Then, after about 60 seconds, click refresh again and the phase Analyze should change to 'Finished' Once you see that it has finished, you can click the Eye icon to see the results:

__177. This will bring up a screen showing the quality scores, data class choices made automatically, and business term assignments made automatically:



In conclusion: what we have accomplished here is that we can understand the quality, possible data classifications, and business terms of a virtual table, in this case containing both structured Db2 data and semi-structured MongoDB data, as we can for any normal table.

The integrated nature of Cloud Pak for Data is what makes it possible to do so many of your tasks from one place.

Lab 14 COGNOS DASHBOARD EMBEDDED - DEEPER DIVE

14.1 Lab overview

When embarking on machine learning projects, many organizations engage Business Analysts to help gain insight into their data. This persona can use tools like Analytics Dashboards (part of the CPD offering) or the more capable cartridge from which this service is derived: Cognos Analytics.

In this lab, you will use the Analytics Dashboards to build visualizations to help the organization understand why their customer use is declining.

In our scenario, the Trade Co. Business Analyst used this service to provide the company executives and the data scientists the information they needed to understand their problem as well as to access the effectiveness of the solution.

14.2 Persona represented in this lab

The Business Analyst persona is likely to perform the exercises in this lab, and that is to create visualizations to make sense of the problems the organization is facing.

Persona (Role)	Capabilities
 Business Analyst	Business Analysts deliver value by taking data, using it to answer questions, and communicating the results to help make better business decisions.

14.3 Logging into the CPD web client (if you have not already done so)

__1. If you are starting this lab stand-alone (without going through previous labs) do the following:

__2. Click the desktop icon: Cloud Pak for Data Web Client.

__3. The CPD web client GUI displays as shown. Use cpduser and cpduser for the *Username* and *Password* and click Sign In.

14.4 Reviewing the dashboard: Monthly Metrics - Trade Co.

In the dashboard Monthly Metrics – Trade Co., the business analyst for Trading Co. has painted a picture of how the business was initially doing. The dashboard shows a relatively flat shares sold per month, and a declining number of traders per month, which the business analyst provides to the Trade Co. executives.

14.5 Building the dashboard: Monthly Metrics - Trade Co.

The business analyst began by analyzing current trends of customer visits and daily trades in the Trade Co. Stock Trader application. He requested the data engineers to provide a file with historical totals of visits and trades for the past year (TraderData.csv) which was deposited into the project where everyone on the team could collaborate.

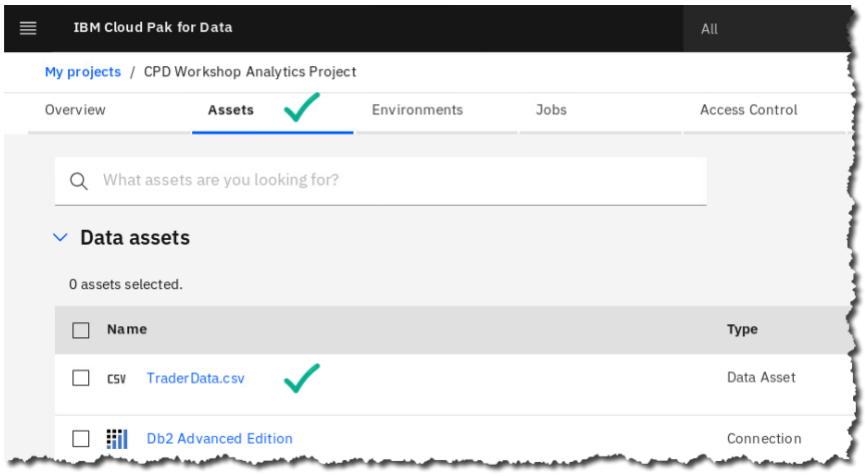
Build the dashboard with that data to see the trends the business analyst discovered.

14.5.1 Starting the dashboard

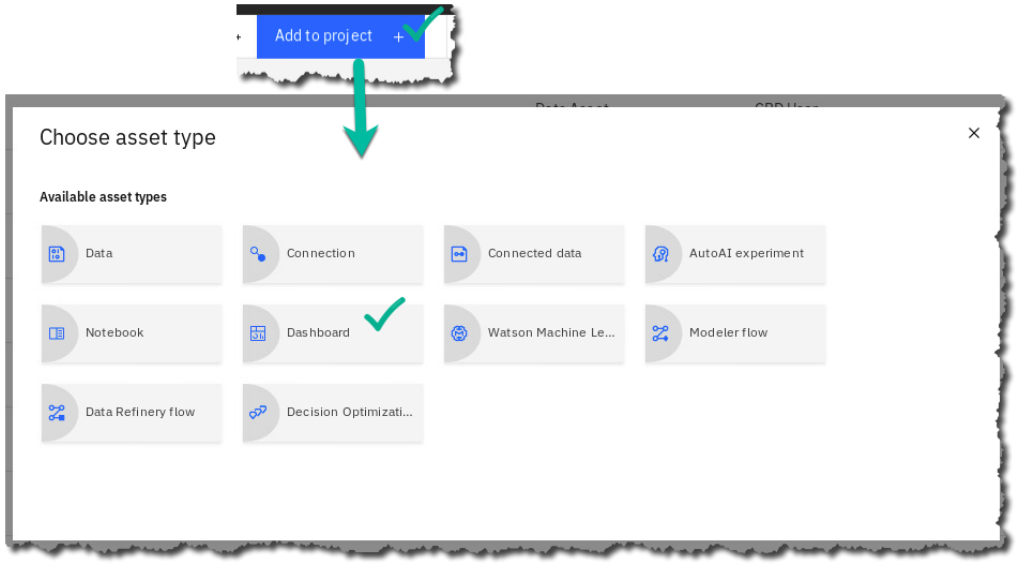
__1. From the top left Navigation Menu ⇨ Projects ⇨ CPD Workshop Analytics Project.

__2. Under Assets click Data Assets.

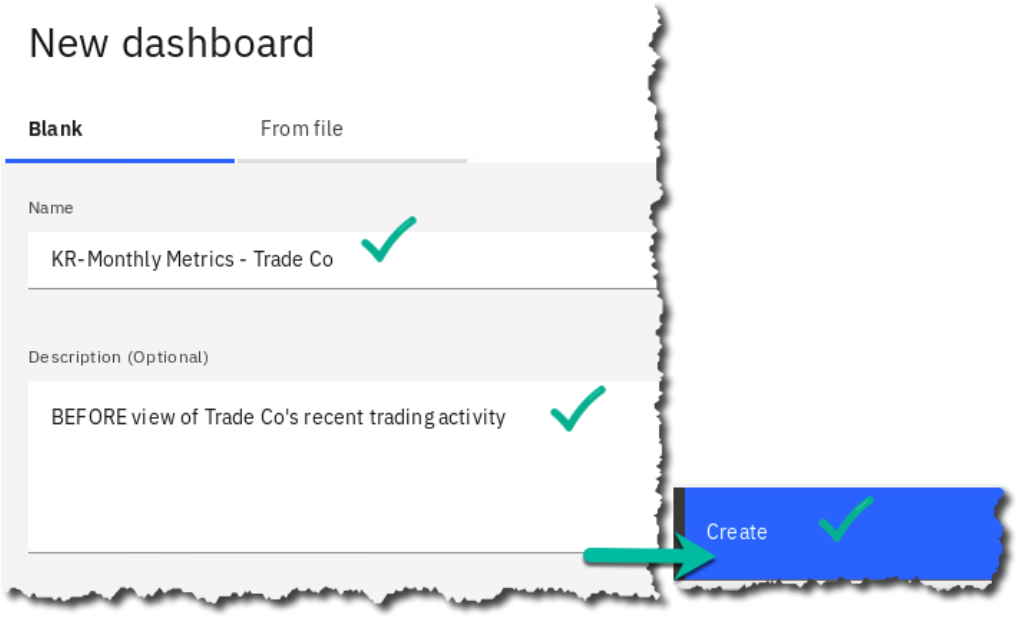
Notice TraderData.csv available in our project.



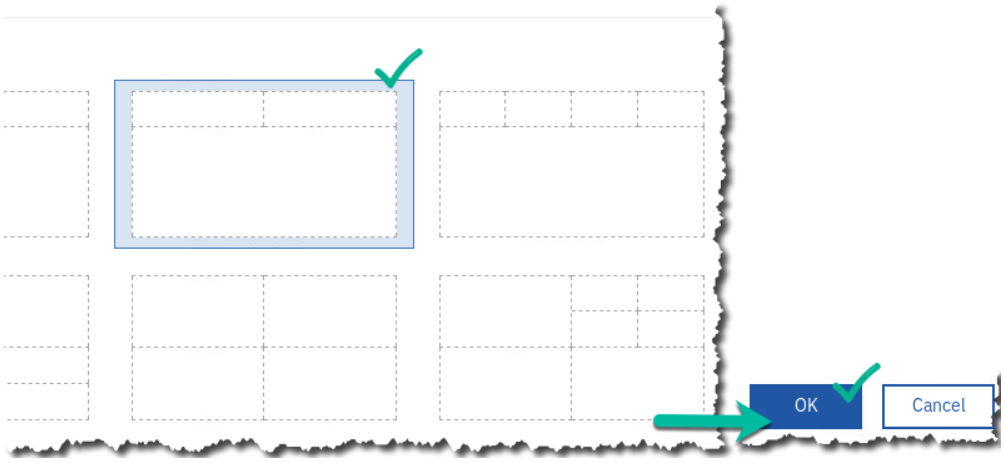
__3. Click Add to project + ⇨ Dashboard.



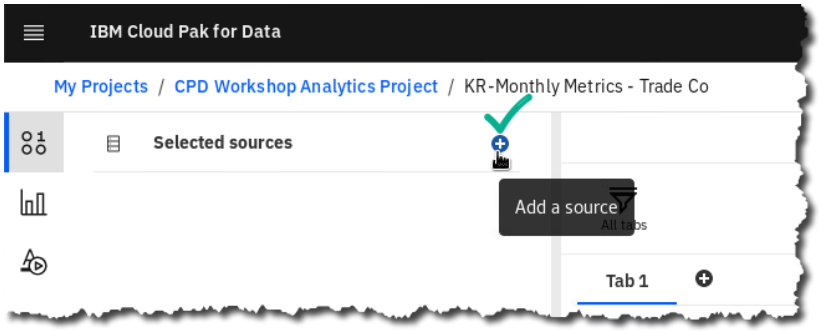
__4. In the Blank tab, fill in the Name and Description as shown below, then click Create - for the name, use initials-Monthly Metrics – Trade Co.



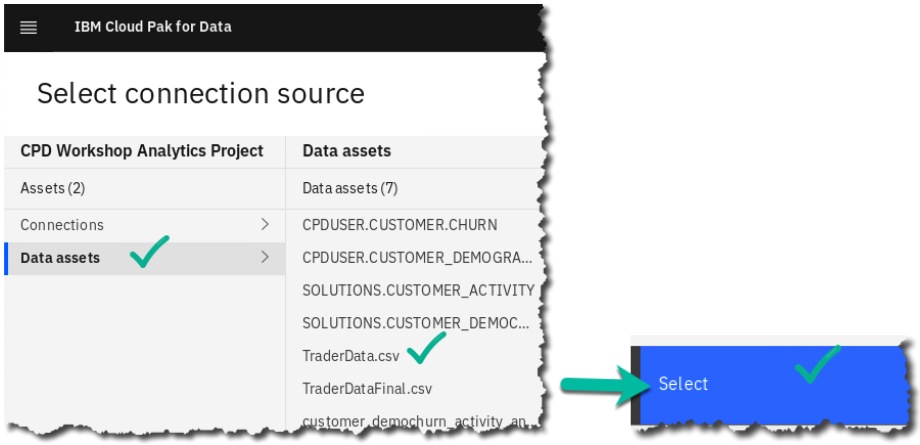
__5. You are now presented with a choice of canvas templates. Select the one that looks as shown ⇨ Click OK.



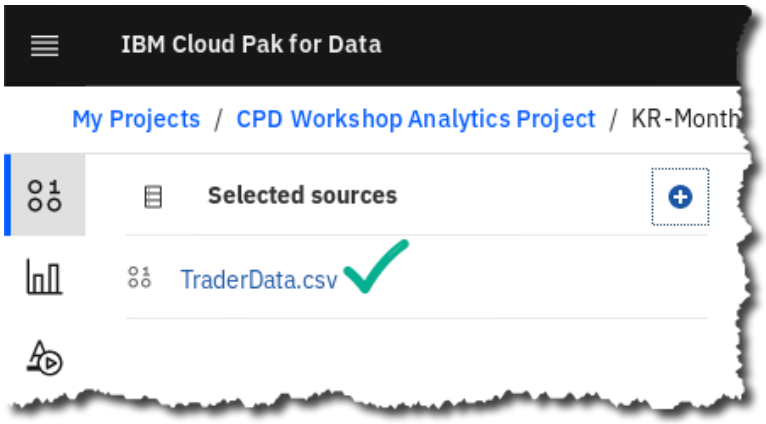
__6. From the Selected sources area near the top left of your screen, click ⊕.



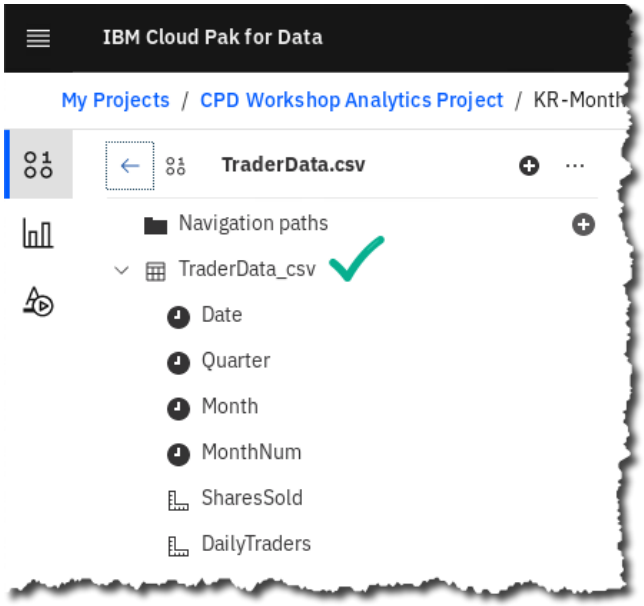
__7. Expand arrow next to Data assets and choose: TraderData.csv ⇨ Select.



__8. Click on TraderData.csv once it is in the Selected sources. This will allow you to expand it.

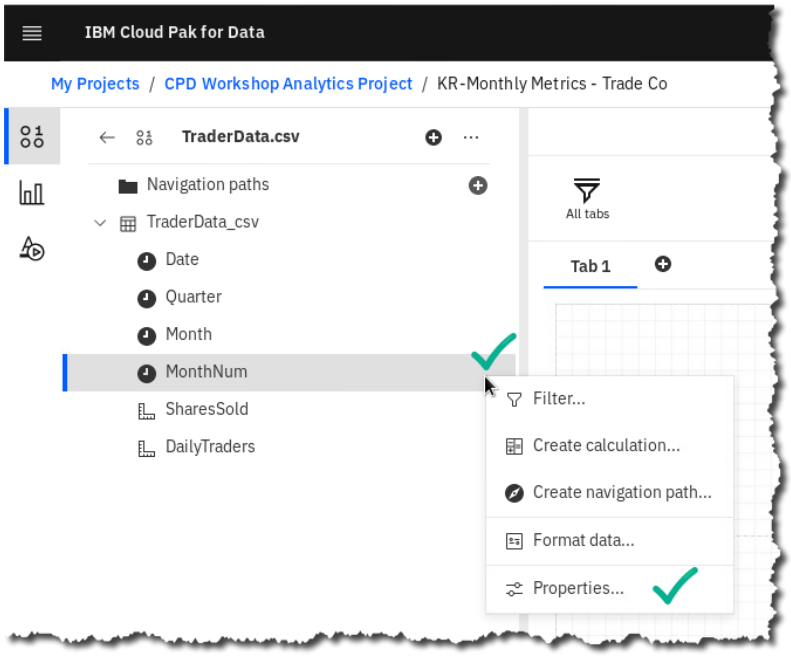


__9. Now expand TraderData.csv.

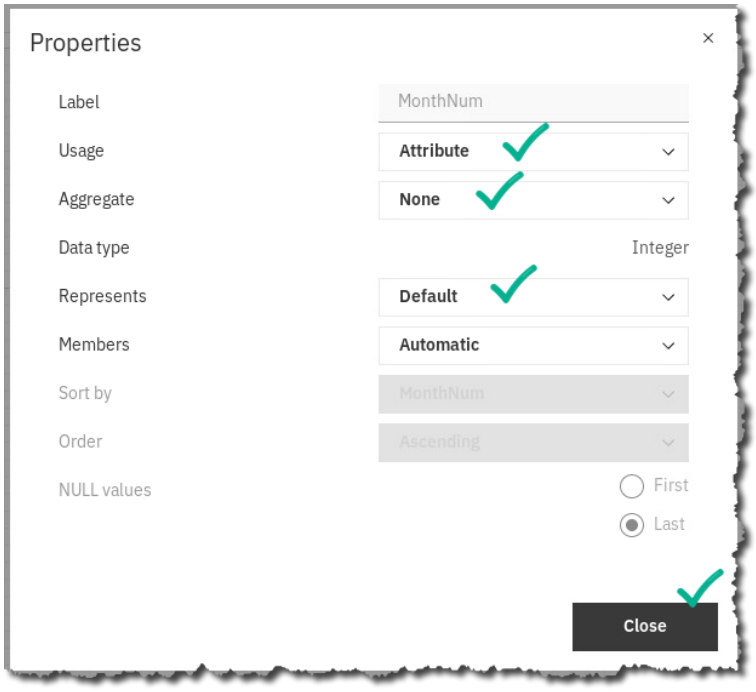


__10. Next. ensure the properties of the column data are what you want to be represented correctly within the dashboard.

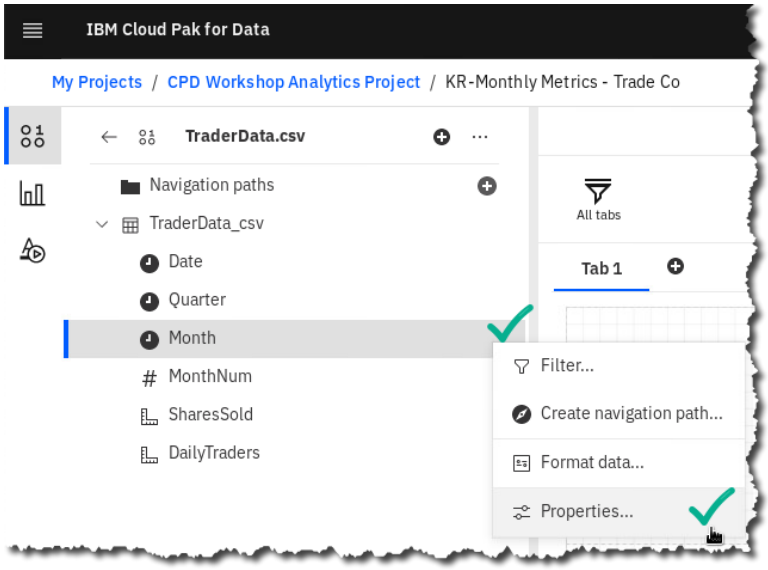
Click the MonthNum ellipsis, then select Properties from the flyout menu.



__11. MonthNum is an attribute on which we sort our months. Change MonthNum usage to be Attribute and Aggregate to be None, Represents Default, then click Close.



__12. Select the Properties of Month by selecting the ellipses next to Month.

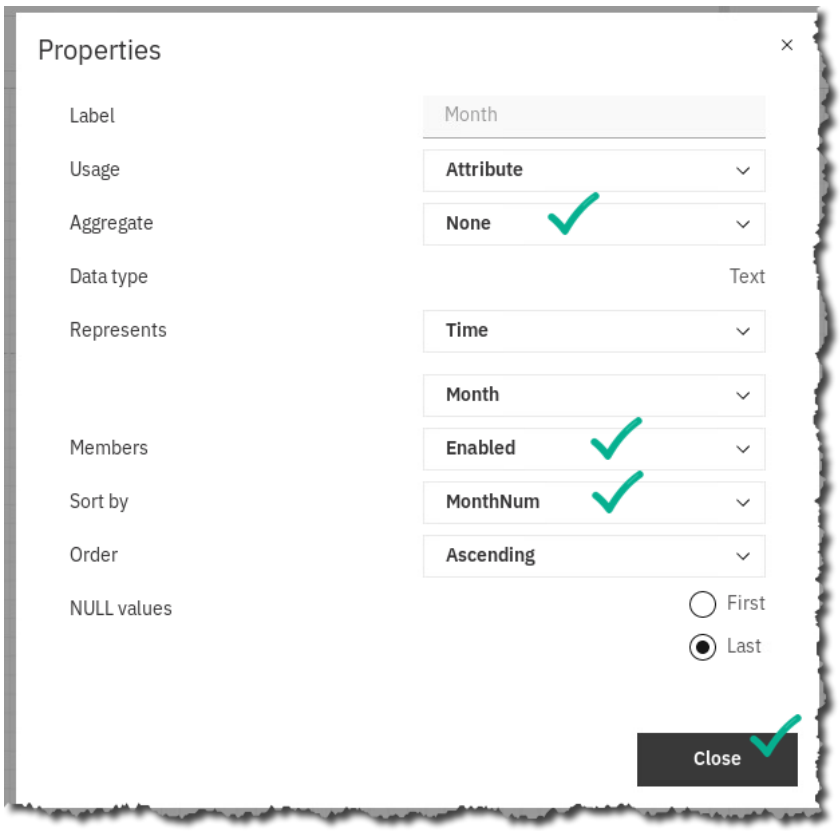


__13. Change Aggregate to None.

Change Members to Enabled.

Select Sort by – MonthNum.

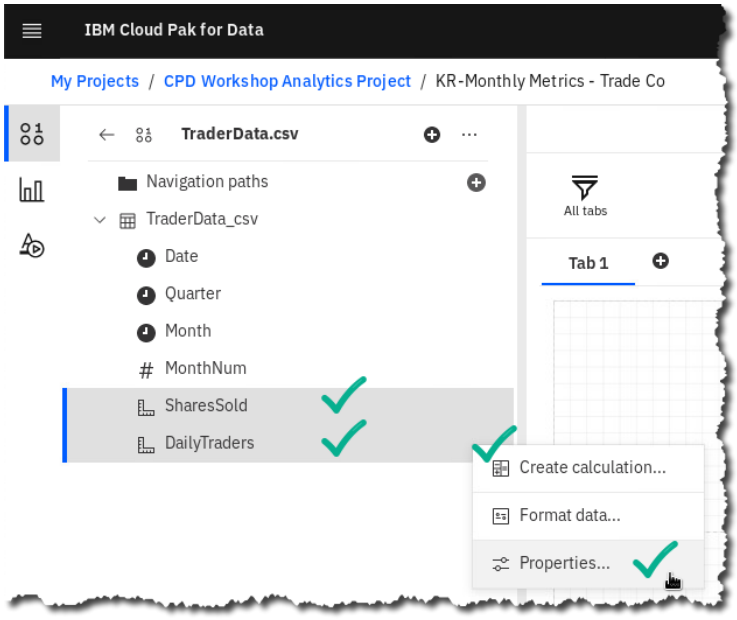
Click Close.



__14. Next, set the aggregation of our measures for totaling.

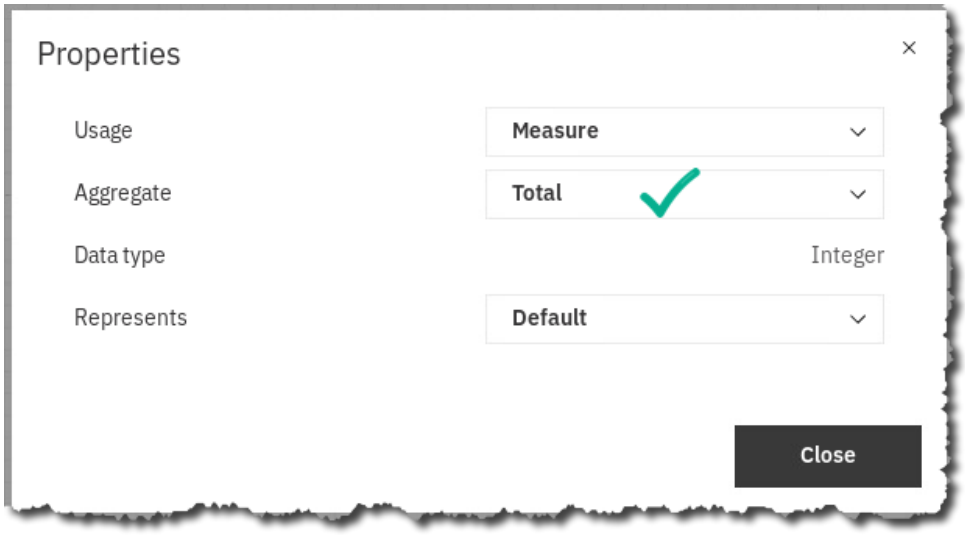
Hold the shift key down and click SharesSold and DailyTraders to select both.

Click the ellipsis and then click Properties.



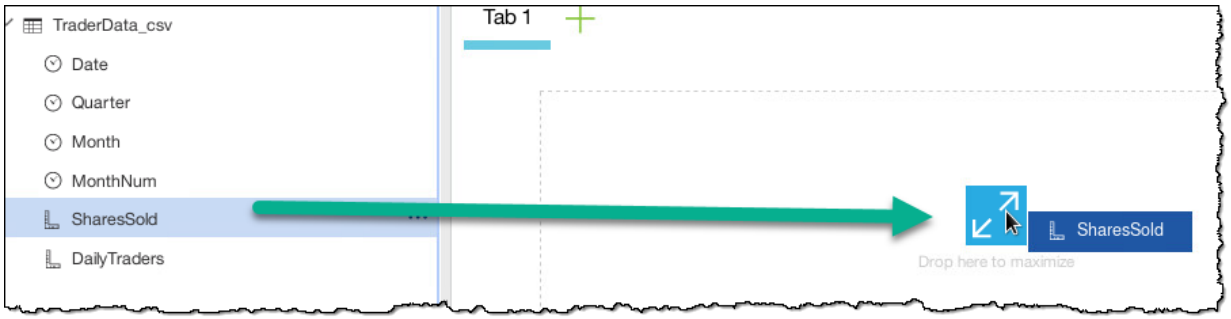
__15. Make sure *Aggregate* is Total in the drop-down menu.

Click Close.



14.5.2 Building the first visualizations

__16. Drag SharesSold to the top left box, hovering over the *Drop here to maximize* area when it turns blue, then releasing the mouse. This gives us a total of Shares sold over all time.



__17. If you did not drop at the right place it won't fill the template area. You can still adjust the guide to match with the outline of the left box. It should look as below.

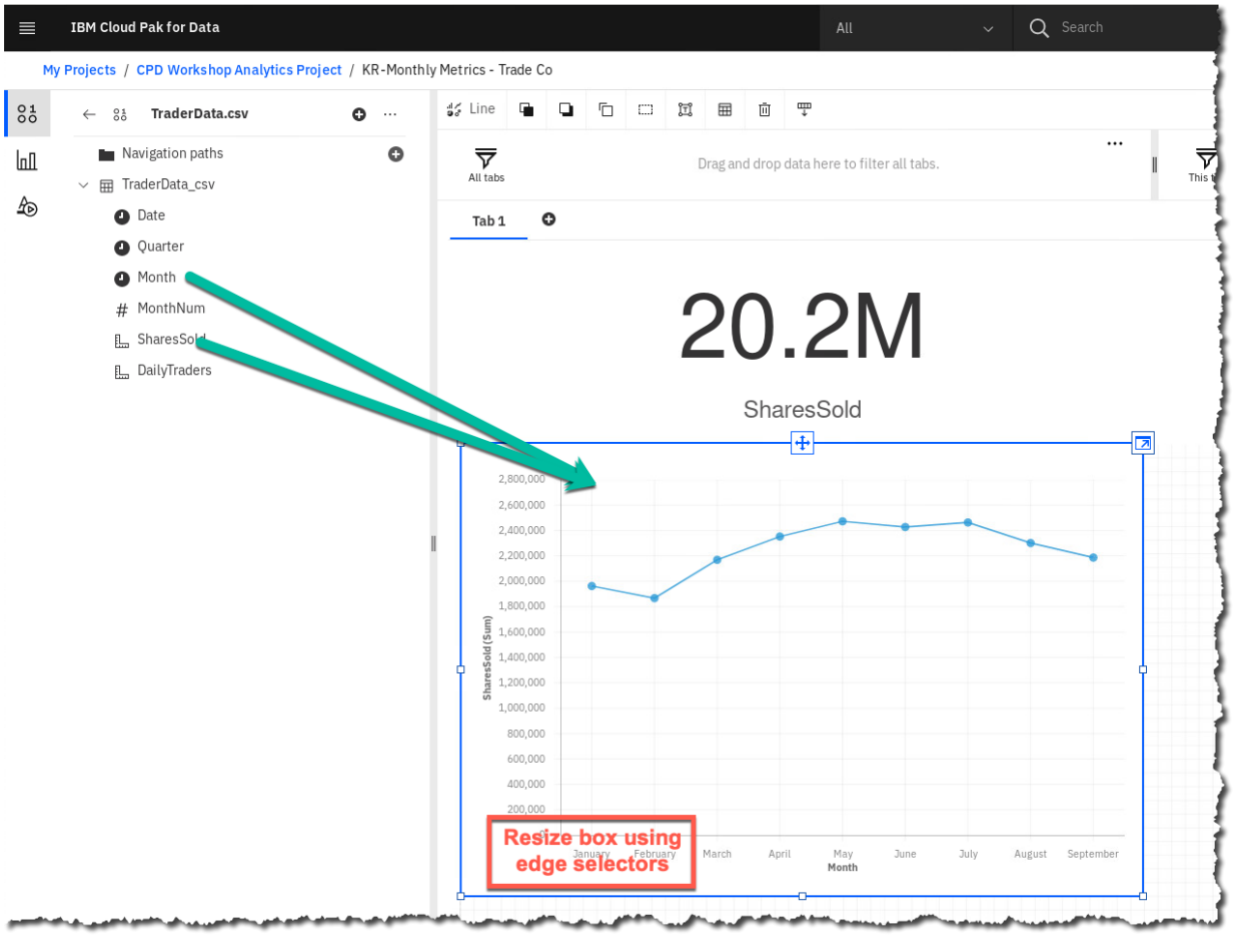
__18. Drag DailyTraders to the top right box, hovering over the *Drop here to maximize* area when it turns blue as well. This maximizes this metric in this box. This gives us a total of trades over all time.

__19. After you complete both top boxes, the dashboard should display as shown:

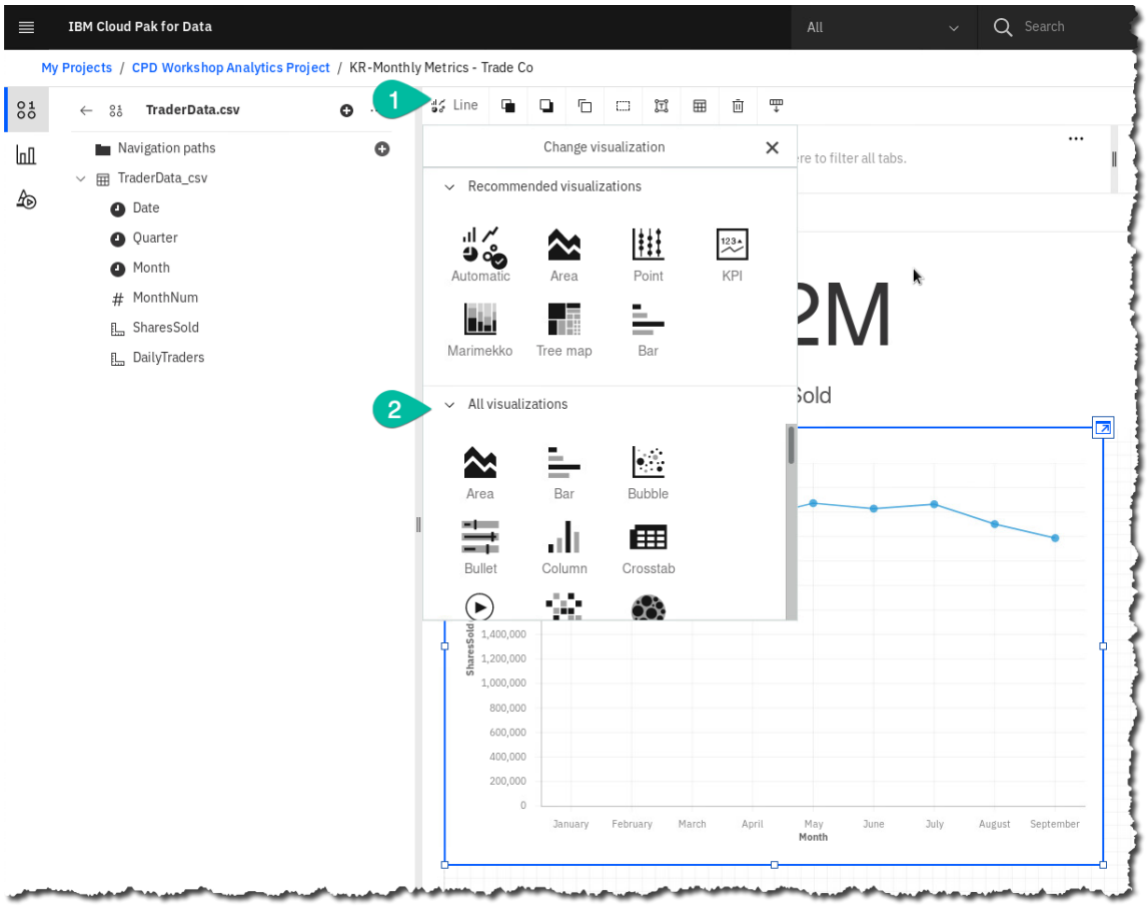
14.5.3 Building the line charts

__20. Hold the [Ctrl] key down *(or Control-Option on Mac),* click Month and SharesSold and drag the two onto the lower left canvas area. This time do not drop in the *Drop here to maximize* area.

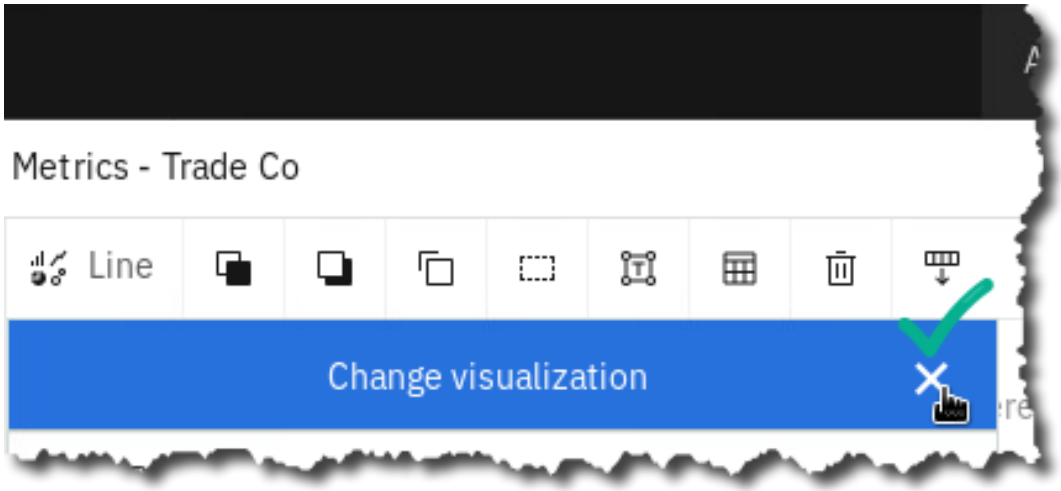
__21. When finished, it will automatically format like this. (Stretch the visualization wider if needed.)



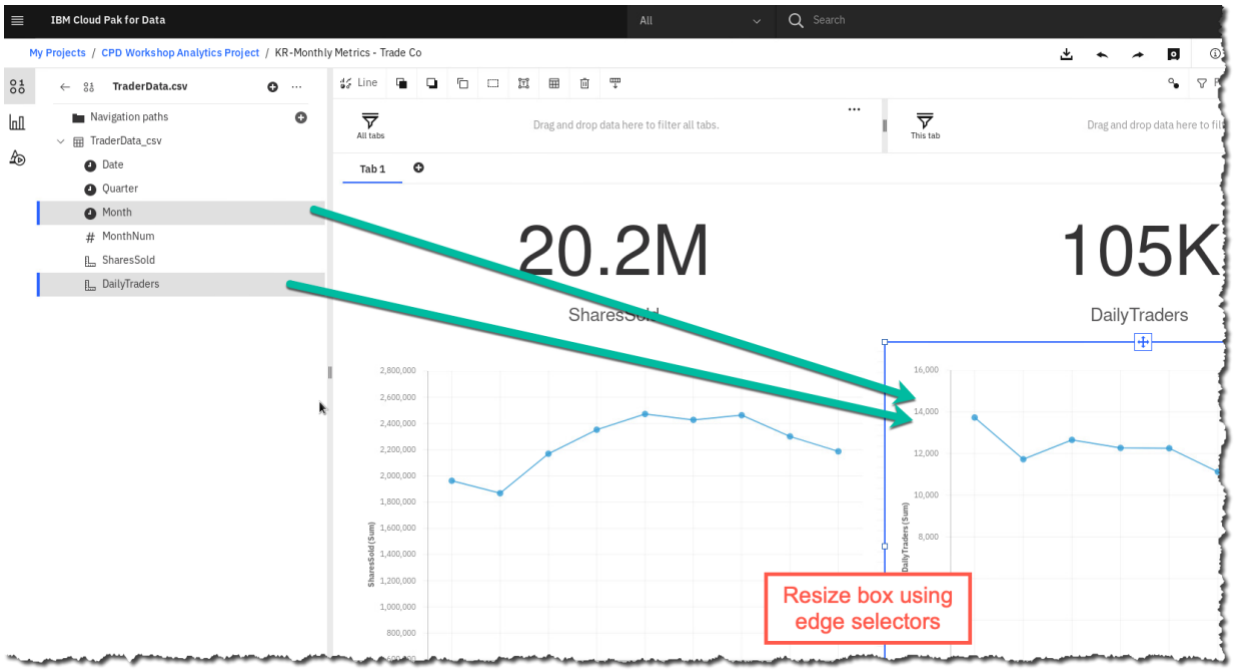
__22. Cognos Analytics uses augmented intelligence to automatically provide you with a chart and appearance that would most likely be represented by the data you select from your data source. You are not limited by this and can select other charts and graphs to represent your data. To see all the available visualizations available at your fingertips, simply select the chart selector (Line, in this case), and view all the visualizations available.



__23. Click the X to close the Change visualization screen.

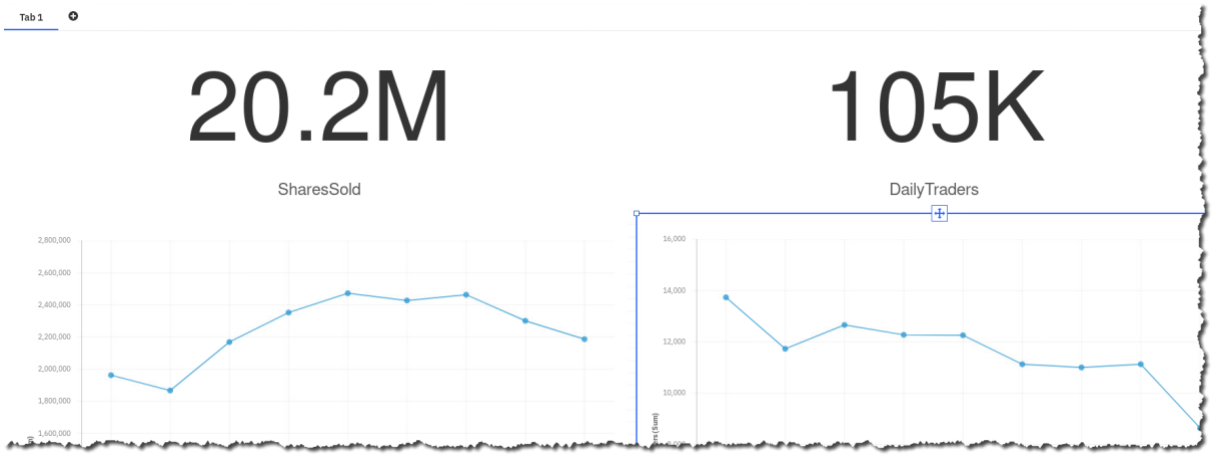


__24. Hold Control key down, click Month and DailyTraders and drag the two onto the lower canvas area. Again, do not drop in the *Drop here to maximize* area.



__25. Adjust the top, bottom, left and right of the chart boundaries so that all the boxes are aligned.

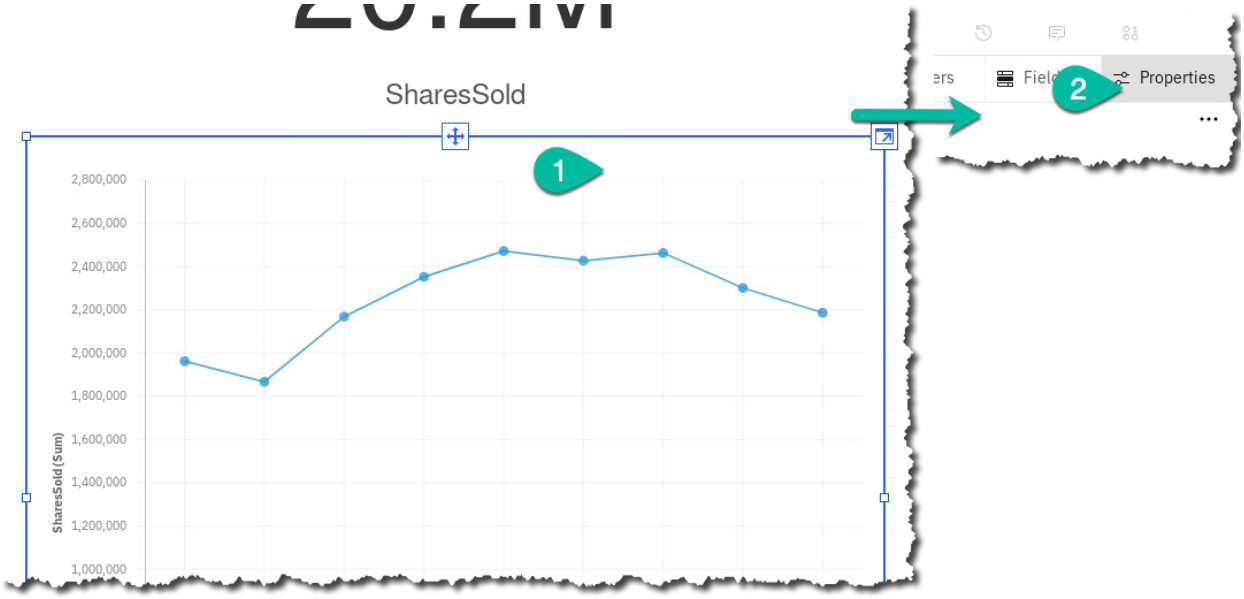
__26. The dashboard should display as below.



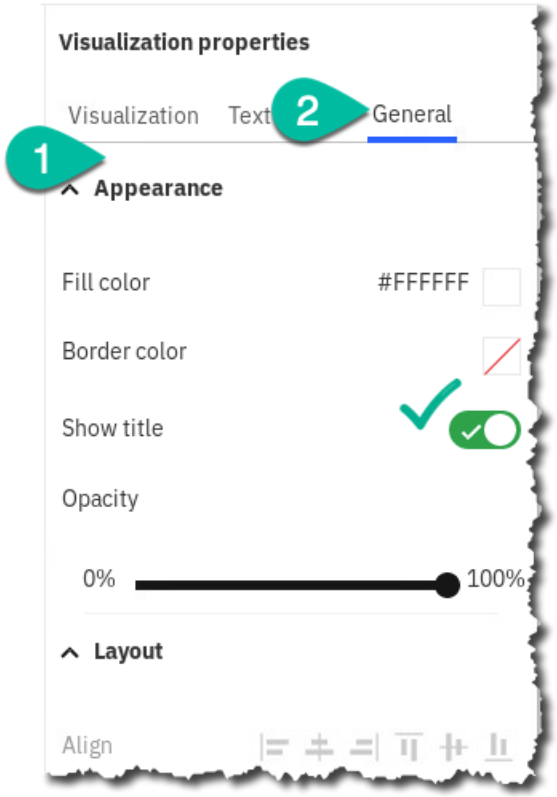
__27. Click the Save icon at the top of the screen to save your work.

14.5.4 Making the finishing touches

__28. Select (by clicking on) the bottom left chart visualization, then select the Properties button at the top of the screen to format that visualization.

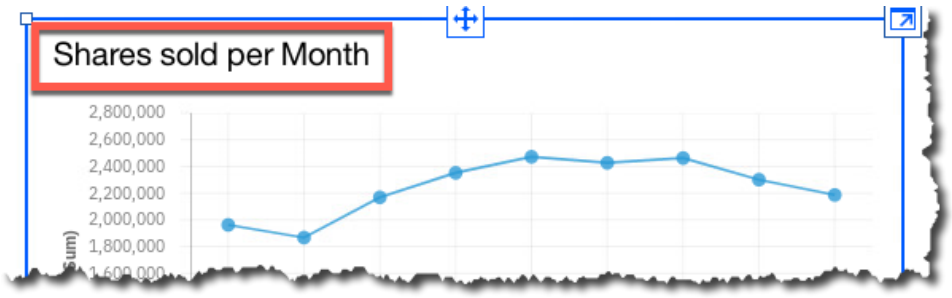


__29. Expand Appearance, then from the General tab, check Show title.

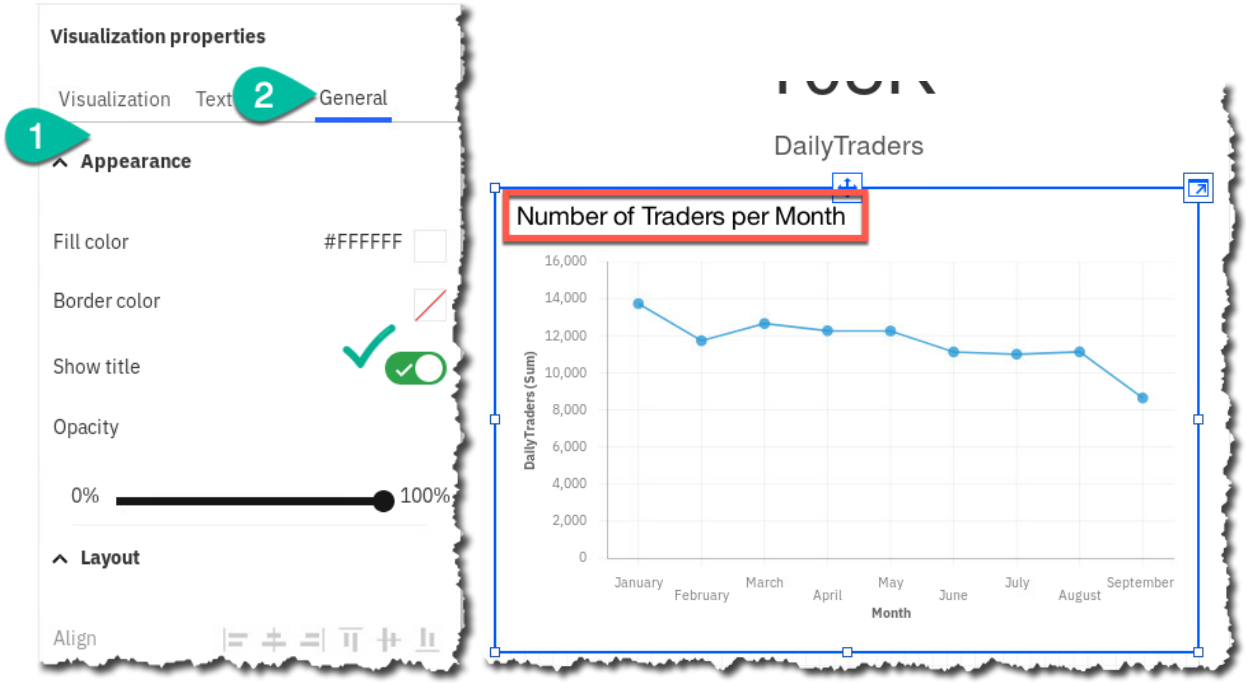


__30. In the bottom left visualization itself you can now type in a title for it.

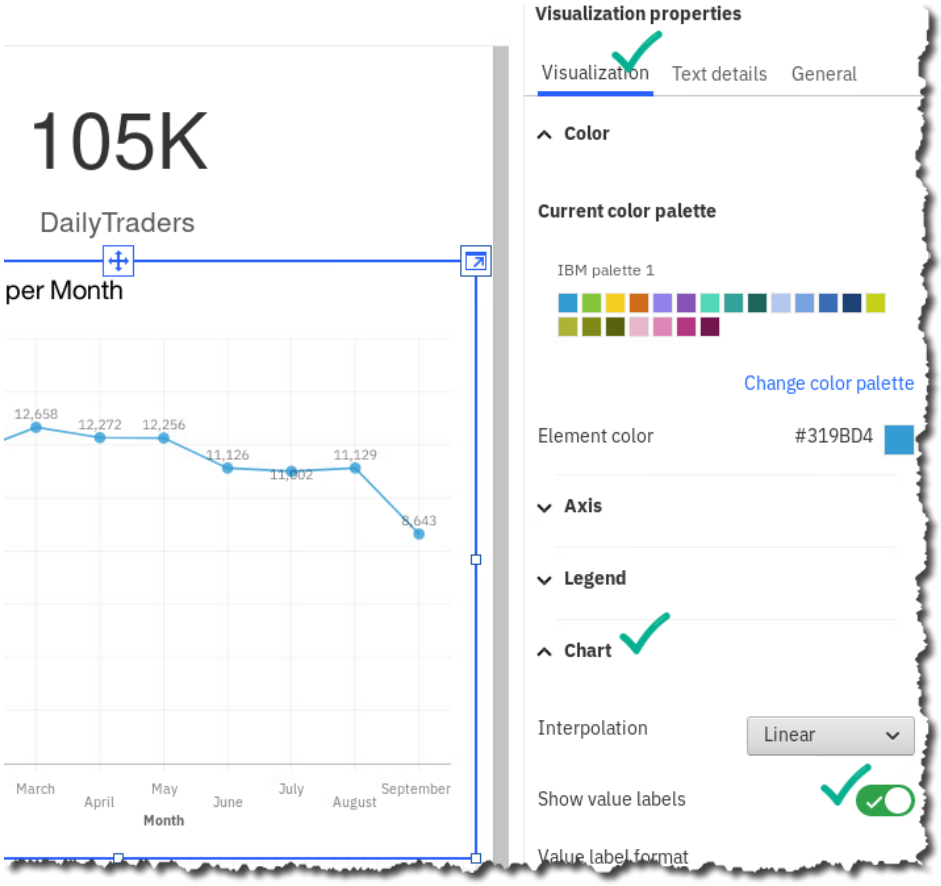
Enter Shares sold per month.



__31. Click the bottom right visualization, select Properties *(if not selected)*, Appearance, General tab, Show title, and enter Number of Traders per month.



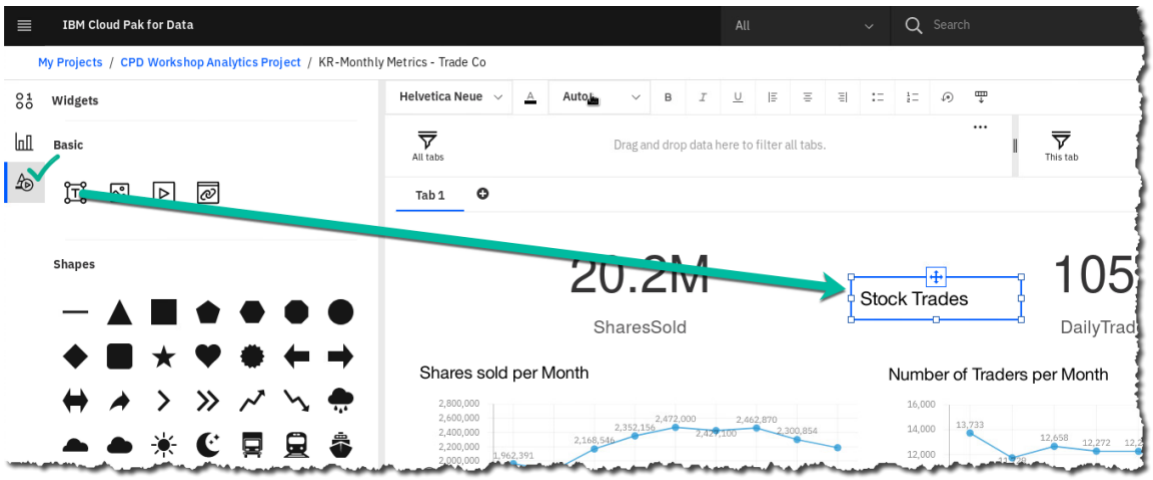
__32. From the Visualization tab, expand Chart, then check Show Value Labels.



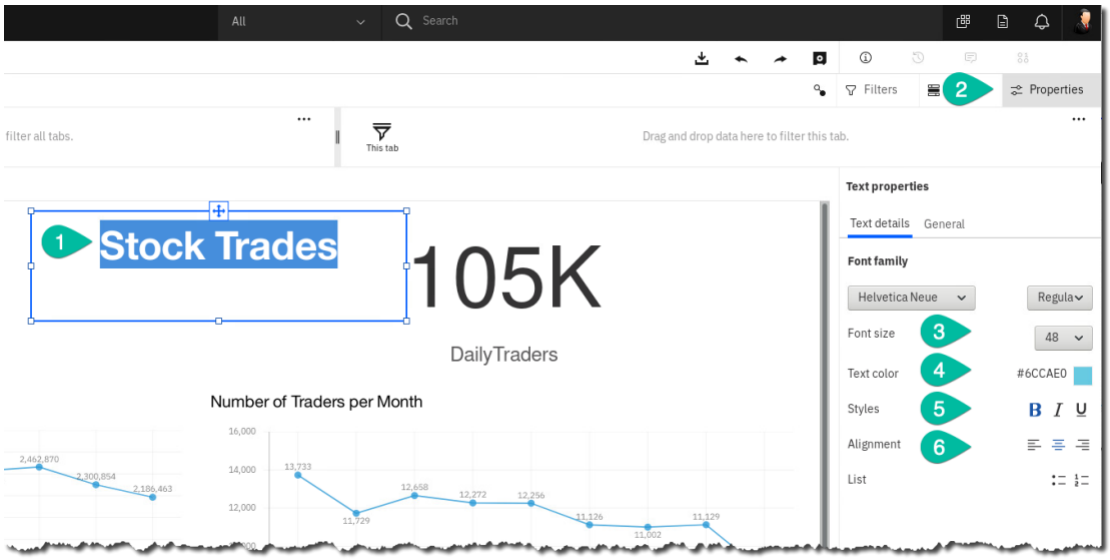
__33. Select the Shares sold per Month chart then check Show Value Labels.

__34. On the left side of the canvas, click the Widgets menu and drag a Text box between the top two visualizations.

Title it Stock Trades. Adjust the box so the title is large and readable.

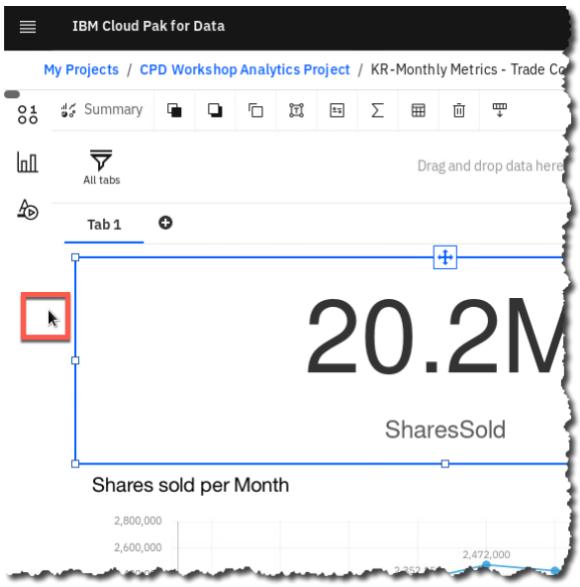


__35. Format the properties of the title Stock Trades. Make the text font size 48, bold, choose light teal text color and center aligned.



__36. Close the Widgets panel.

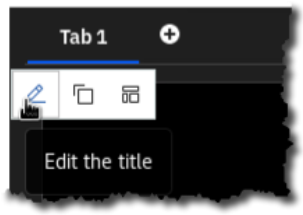
__37. Next, let's change the overall look of the dashboard to match our desired look and feel. Select somewhere outside the boxes to select the overall dashboard itself. For example, click to the left of the top left visualization as shown:



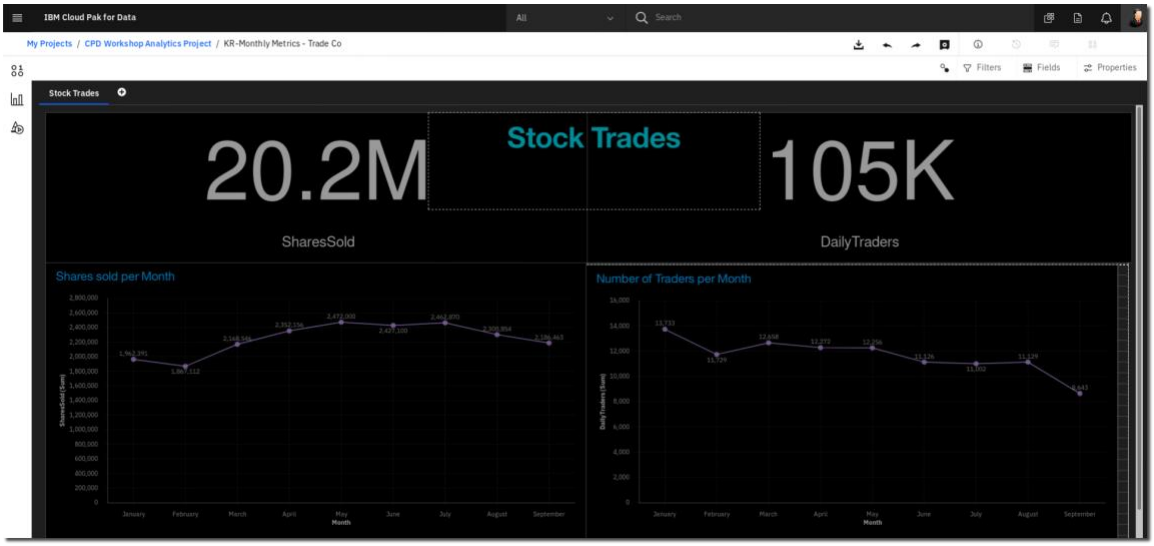
__38. This selects the entire dashboard. Select Properties again, expand Color and theme and choose Dark.

__39. Close Properties by selecting Properties.

__40. Select tab at top left and choose the Pencil icon to Edit the title. Give it a title of Stock Trades.



__41. The final results should look something like this:



__42. Click Save again to mark your progress.

You will notice from examining the charts that Shares sold is relatively flat and daily trades are falling off. We need to use CPD to discover the WHY behind this trend.

14.6 Lab conclusion

Dashboard creation and analysis is just one part of the Analyze phase, but a business analyst can do this relatively quickly and simply so that they can get information from your organization's data without relying on data scientists for everything.

However, the Data Scientist can take this information and create machine learning models from the knowledge gained by this kind of analysis.

**** End of Lab 14 - Cognos Dashboard Embedded – Deeper Dive**

Lab by Burt Vialpando and Kent Rubin

Lab 15 SPSS USING NPS – DEEPER DIVE

15.1 Lab overview

In this lab you will use SPSS Modeler (premium cartridge) to build a churn model using training data stored in Netezza Performance Server (NPS). You will also use a Jupyter notebook to build a churn model using training data stored in NPS.

These two methods for building models are distinctly different: SPSS uses a clickable paradigm, where Jupyter notebooks use coding. After building the models you will deploy them in a Cloud Pak for Data deployment space.

You will then create two types of deployments: batch and online. Finally, you will execute the deployments to complete an end-to-end example of training, deploying and using predictive models via the Cloud Pak for Data platform with NPS data.

15.2 Persona represented in this lab

The `Data Scientist` persona is the likely role to perform all the tasks shown in this lab.

Persona (Role)	Capabilities
 Data Scientist	Data Scientists bring expertise in statistics and the process of building ML/AI models to make predictions and answer key business questions.

15.3 Logging into the CPD web client (if you have not already done so)

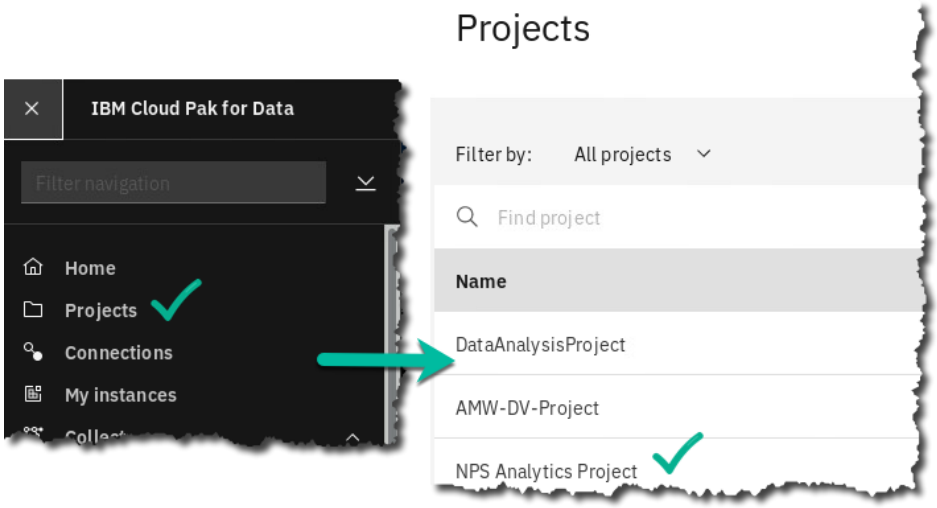
__1. If you are starting this lab stand-alone (without going through previous labs) do the following:

__2. Click the desktop icon: `Cloud Pak for Data Web Client.`

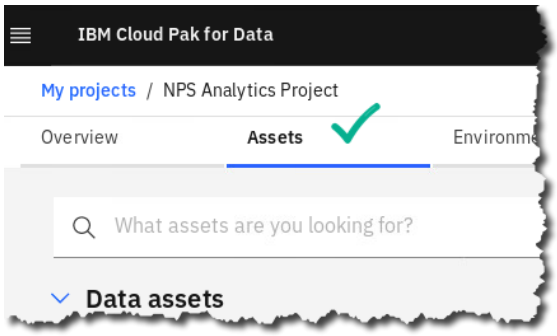
__3. The CPD web client GUI displays as shown. Use `cpduser` and `cpduser` for the *Username* and *Password* and click `Sign In`.

15.4 Working with NPS Data in SPSS

__4. Click `Navigation Menu` ⇨ `Projects` ⇨ `NPS Analytics Project.`



__5. Click `Assets.`



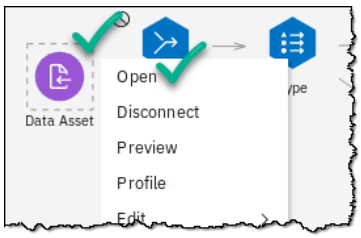
__6. Scroll down to the Modeler flows Section and click `Churn Prediction Flow.`

 Data Scientist	Note: This SPSS Modeler flow performs data preparation, model building, and model evaluation, all without requiring coding, by using a graphical user interface.

15.4.1 Importing data

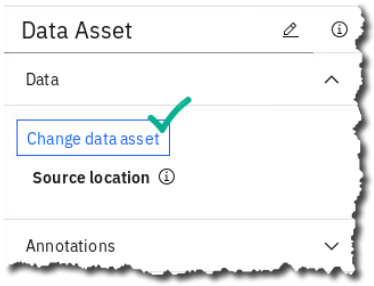
__7. You'll need to connect to three tables stored in NPS. To connect to the first table, do the following:

Mouse over the `Data Asset Import Node` ⇨ `Right click` (or click the `ellipsis)` ⇨ `Open.`

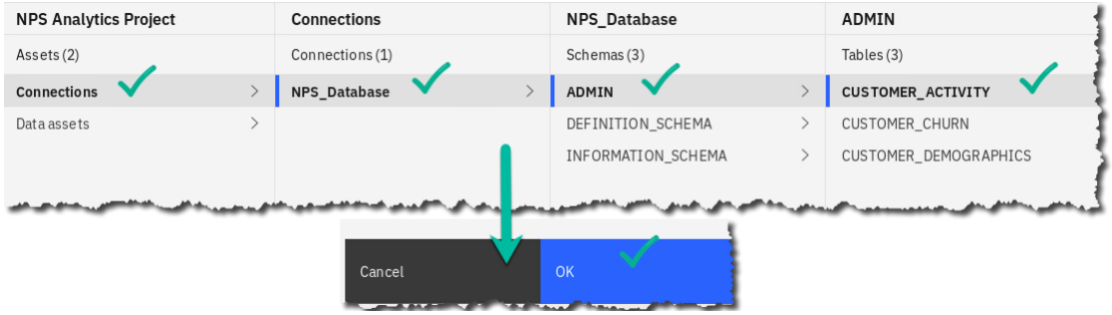


__8. On the right side of the screen a pop-up will appear.

Select `Change data asset`.

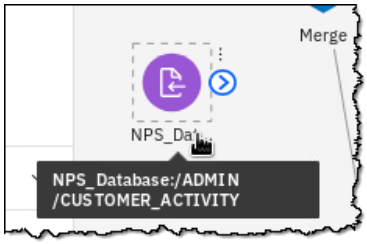


__9. Click `Connections` ⇨ `NPS_Database` ⇨ `ADMIN` ⇨ `CUSTOMER_ACTIVTY` ⇨ `OK.`



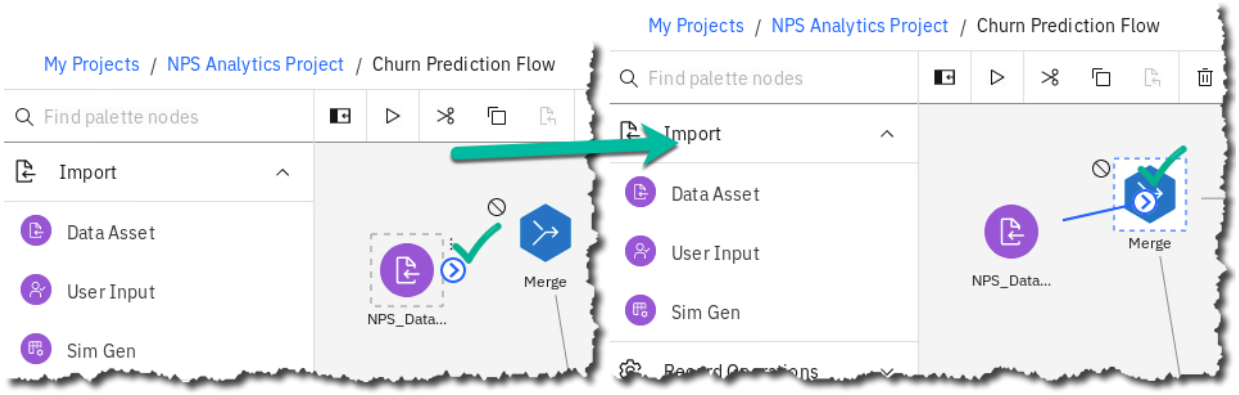
__10. Click `Save` in the lower right to finish setting up the Data Asset Import node to the first table in NPS.

__11. Hover over the node on the canvas and notice it has changed its name to `NPS_Database:/ADMIN/CUSTOMER_ACTIVITY.`

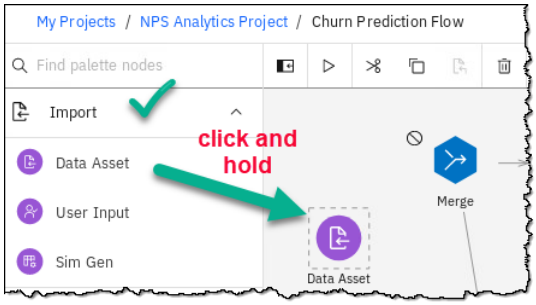


__12. Mouse over the `Data Asset Import` node you are working with until an arrow in a circle appears.

Click and hold the arrow while dragging to the `Merge` node located just to the right. Release the mouse button to connect the two nodes.

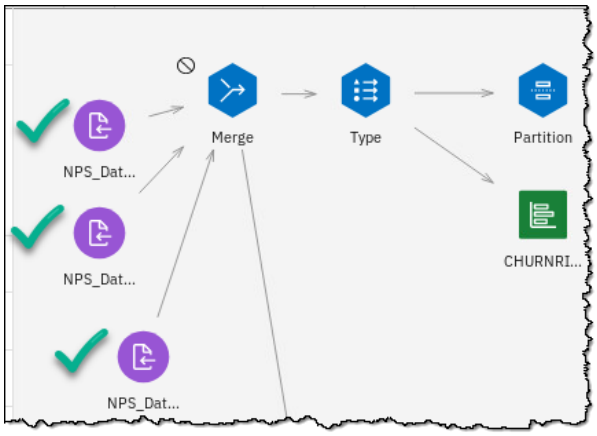


__13. Now create and connect to the remaining two tables by clicking the dropdown menu `Import`, then click and hold `Data Asset` and drag and drop it onto the canvas to the right. Follow the `Importing Data` steps above to create connection for `CUSTOMER_CHURN and CUSTOMER__DEMOGRAPHICS`.



When done, you will have three Data Asset Import nodes connected to these tables in NPS `CUSTOMER_ACTIVTY`, `CUSTOMER_CHURN`, `CUSTOMER_DEMOGRAPHICS`.

Tip: Hover over each node to make sure you have three different tables represented by the three nodes you just created and subsequently merged.

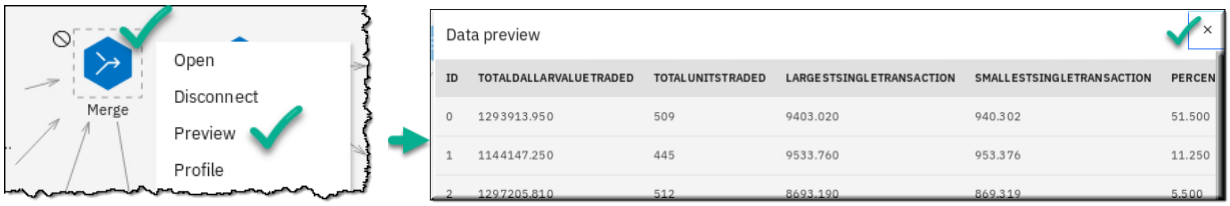


15.4.2 Testing the merged data

__14. To test your work, do a preview of the merged data.

On the `Merge` node click the `ellipsis` ⇨ `Preview`.

At this point you have merged all the training data needed to train and test the churn model.



Click the `X` in the upper right of the `Data preview` to close the window.

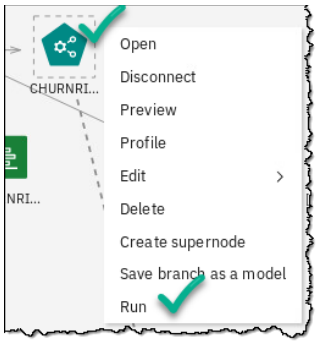
15.5 Building, evaluating, and saving the SPSS model

In this section you will retrain the SPSS model using the training data from NPS. Then, you will evaluate the goodness of the model. Finally, you will deploy the model and save a batch of data for testing the deployed model.

15.5.1 Rebuild the model

__15. Rebuild the model by first finding the `green CHURNRISK` model building node.

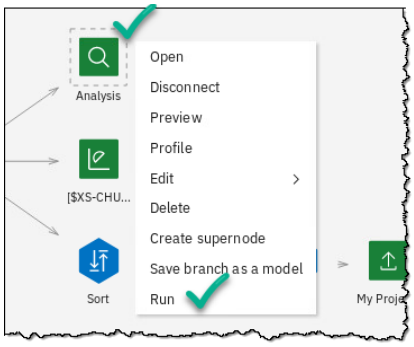
Click `green CHURNRISK (node)` ⇨ `ellipsis` ⇨ `Run`.



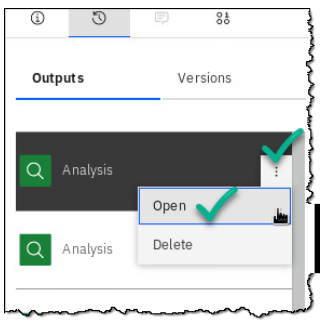
__16. You will see the flow running...

__17. Evaluate the goodness of the model by hovering over the ellipsis on the `Analysis` node.

Click `Analysis (node)` ⇨ `Ellipsis` ⇨ `Run.`

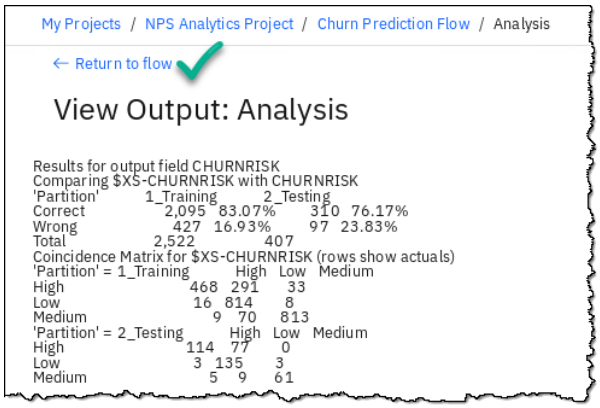


__18. In the pop up box `Outputs` section, click `Analysis` ⇨ `Open.`



__19. Review the information in `View Output: Analysis.`

When finished, click `Return to Flow.`

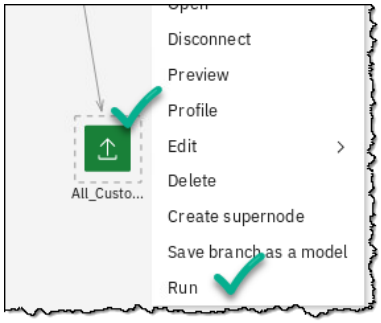


15.5.2 Prepare test data

The portion of an SPSS flow that is to be deployed to Cloud Pak for Data will receive its input data from a CSV file. You will prepare for this by creating a small sample file in CSV format.

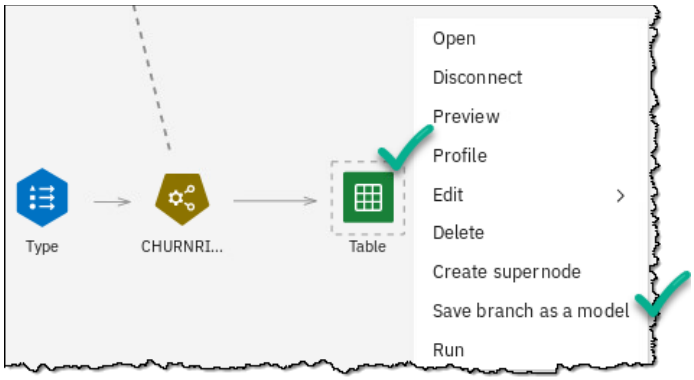
You will save this file for use by the portion of the flow to be deployed as a model.

__20. Find the `All_Customer_Data` node ⇨ Click the `ellipsis` ⇨ `Run`.



__21. The node to the immediate right of the Data Asset Export node that you just ran reads the CSV file you created and provides it as input to a copy of the Churn model you built earlier.

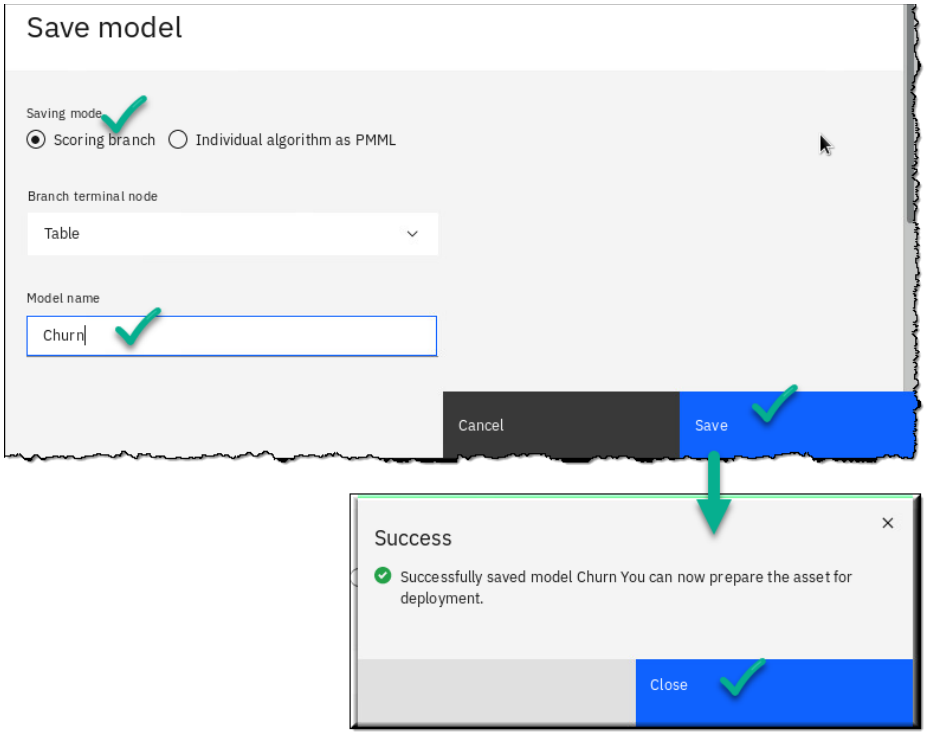
Hover over the `Table` node ⇨ Click on the `ellipsis` ⇨ `Save branch as a model`



__22. Leave the Saving mode as Scoring branch.

Name the model `Churn.`

`Save` ⇨ `Close`.

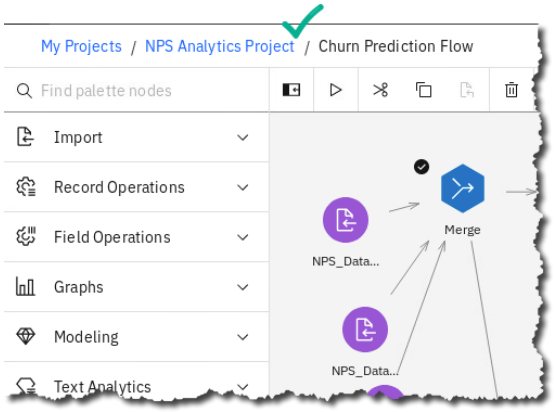


15.6 Creating and testing an online model deployment

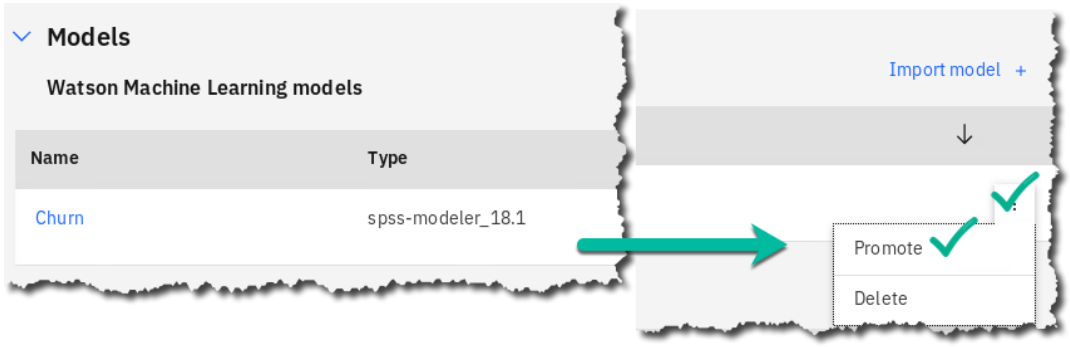
Now you will create and test an online deployment using the model you just saved.

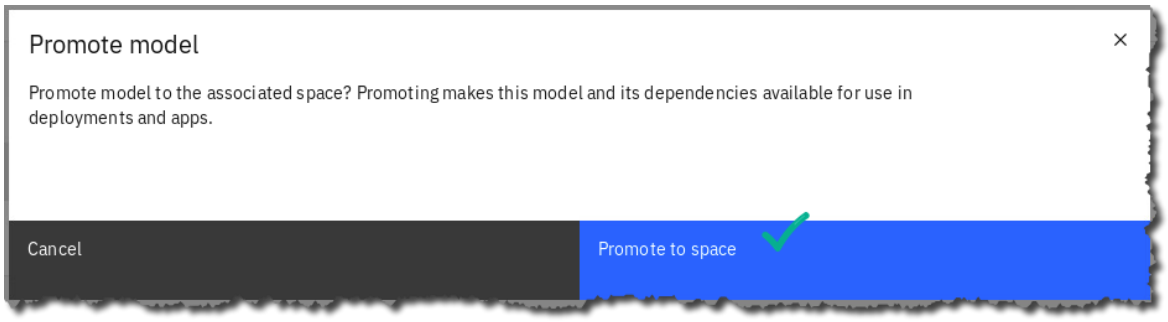
15.6.1 Create the model deployment

__23. Click on `NPS Analytics Project.`

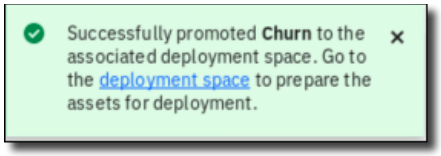


__24. Scroll down to the Models Section and click on the `ellipsis` for the Churn model ⇨ `Promote` ⇨ `Promote to space.`





You will see a message near the top of your screen indicating success.



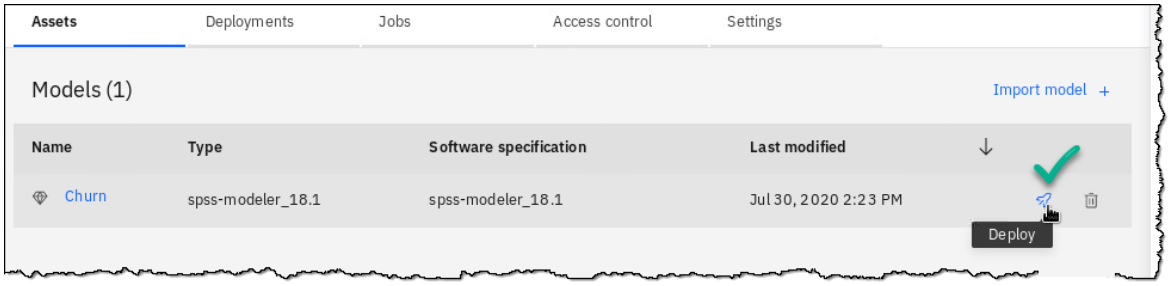
__25. Scroll to the top of the screen and click `Settings.`

__26. Scroll down to find section `Associated deployment space.`

Click `NPS Analytics Deployment Space.`

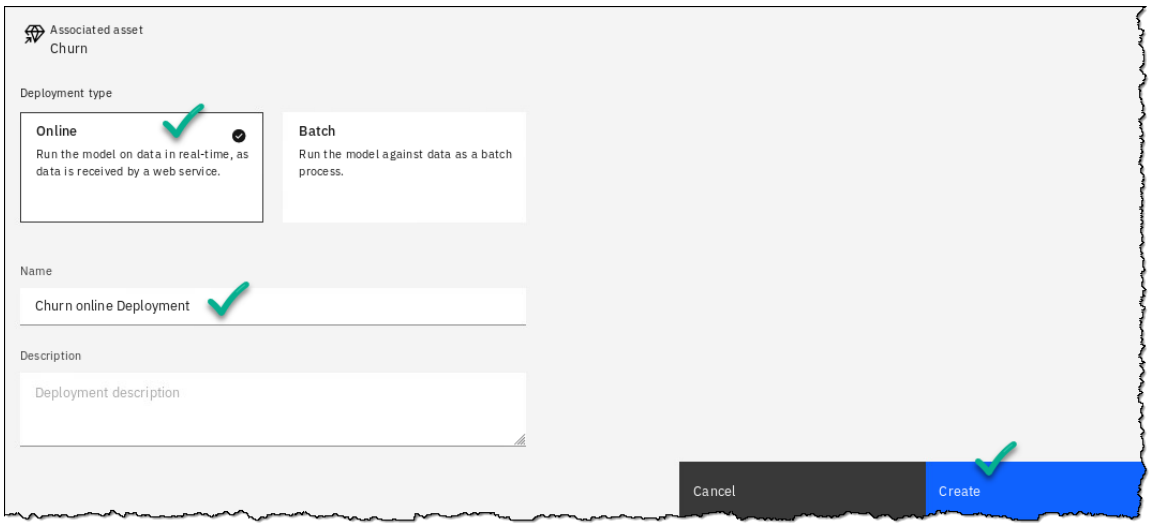
__27. Locate the model `Churn` and `hover` at the end of the line.

Click `Deploy`.



__28. Create an online deployment.

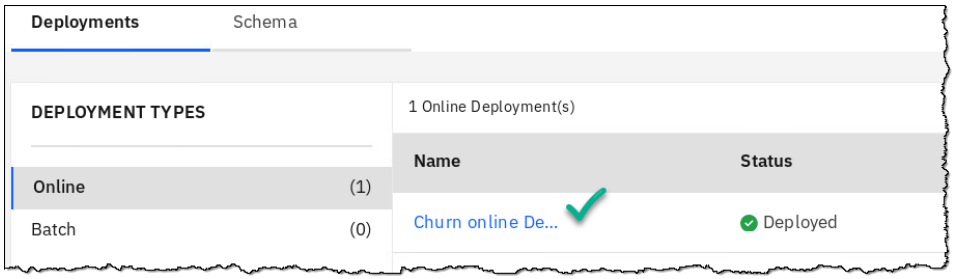
Select `Online` ⇨ Name it `Churn online Deployment` ⇨ `Create`.



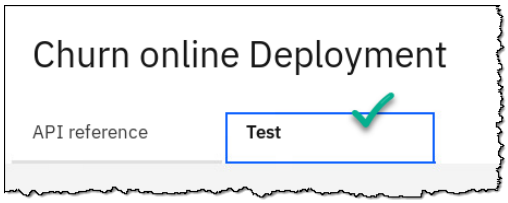
__29. Wait for the Status to change to Deployed.

15.6.2 Test the online model deployment

__30. Click `Churn online Deployment.`

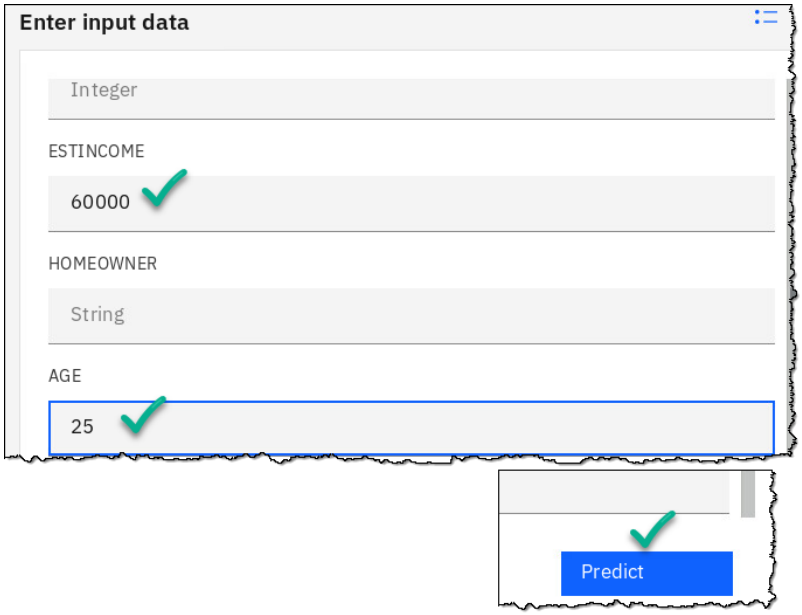


__31. Select tab `Test.`

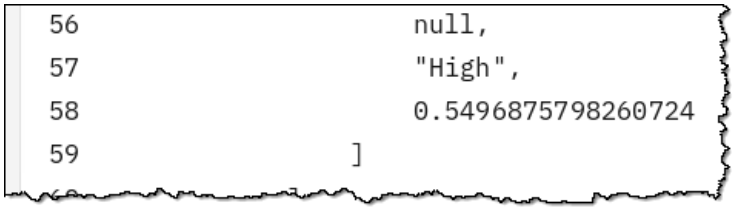


__32. Enter input data by first scrolling down until you can see the `ESTINCOME` and `AGE` fields, enter 60000 for `ESTINCOME` and 25 for `AGE.`

Click `Predict` ⇨



__33. Scroll down in the `Results` window and you'll see the churn prediction along with the confidence in the prediction.



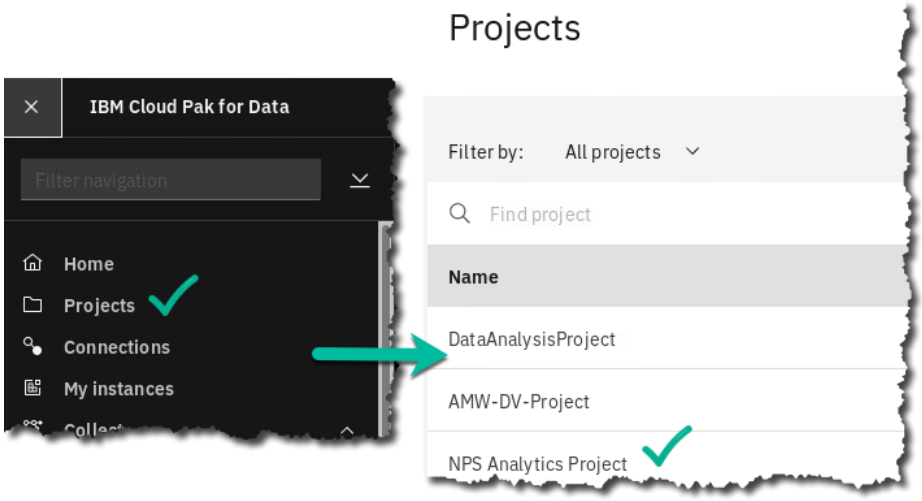
15.7 Working with NPS Data in Jupyter Notebooks

In this section you will use a Jupyter notebook to train and deploy a predictive model. This is a coding approach and this notebook is written in Python.

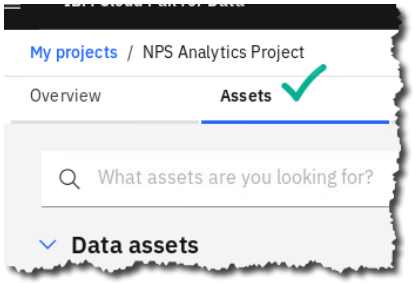
The notebook is complete except for the connections to the three tables containing training data stored in NPS. You need to add the code to accomplish this. However, you'll find that the code will be automatically generated for you.

15.7.1 Launch the notebook

__34. Click `Navigation Menu` ⇨ `Projects` ⇨ `NPS Analytics Project.`

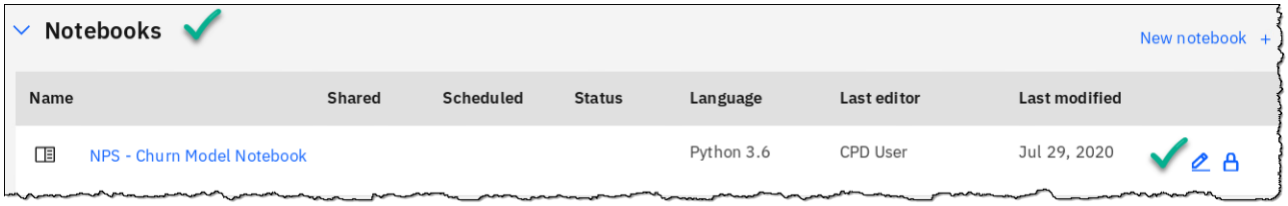


__35. Click `Assets.`

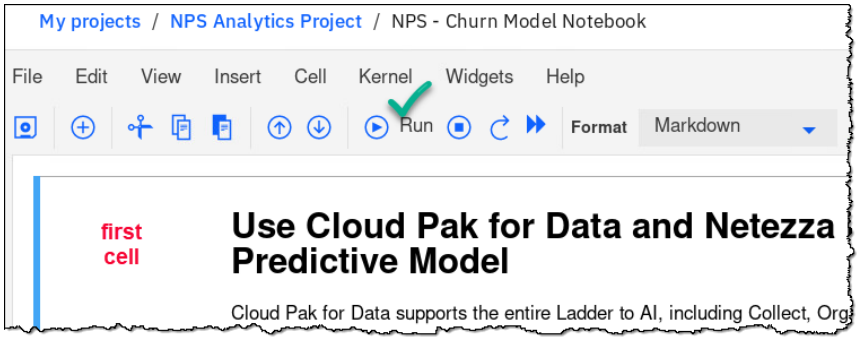


__36. Scroll down to the `Notebooks Section.`

Click the `Pencil icon` next to the Notebook named `NPS - Churn Model Notebook.`



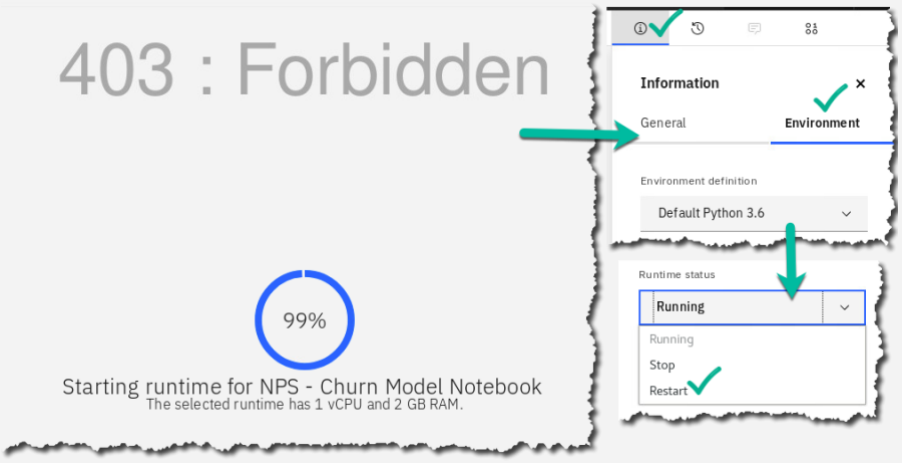
__37. You will start by selecting the first cell by `clicking` on it ⇨ Read the contents of the cell then click `Run.`



You can tell when the notebook is running a cell by observing a solid black circle in the upper right-hand corner of the screen. It turns to an open circle when the notebook has completed its run. After any individual cell has run, any output from the code is displayed immediately after that cell, and the next cell is automatically selected for you.

Data Scientist

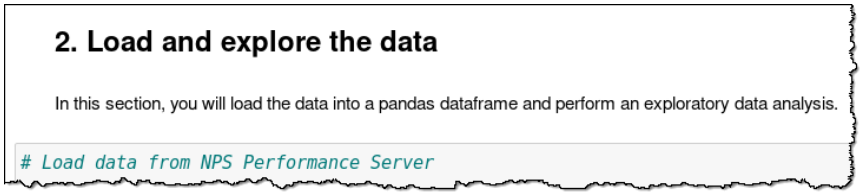
If you receive a 403: Forbidden error restart the notebook by finding the information icon at the top right of the screen and clicking as shown below.



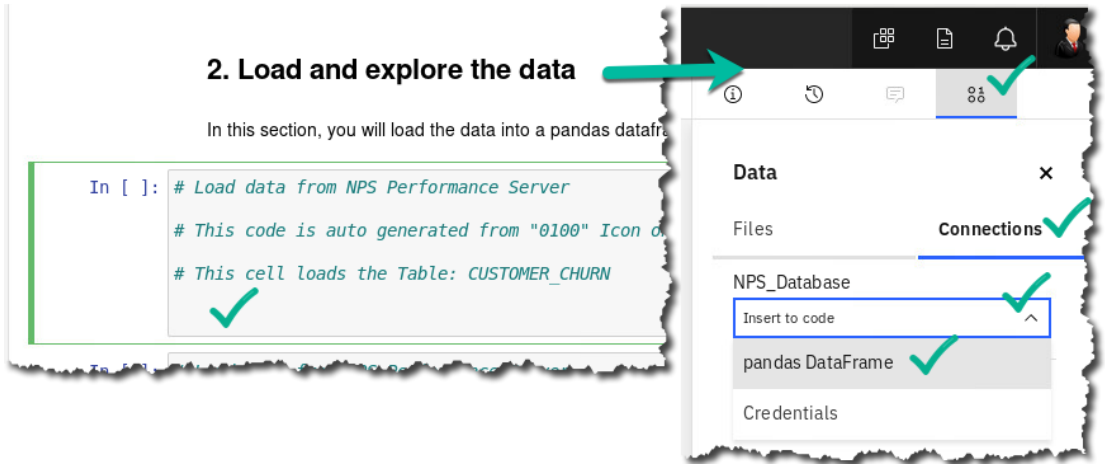
15.7.2 Load data into the notebook

__38. Continue reading and running each cell until you reach the three cells containing the text.

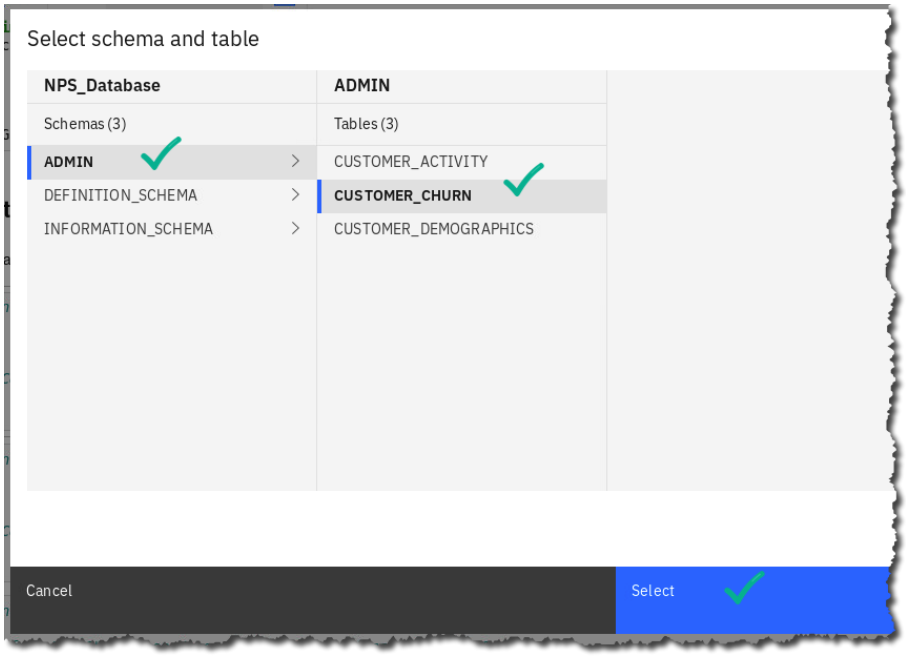
```
# Load data from NPS Performance Server
```



__39. Position and Click your cursor on the last blank line inside the first of the three data cells under the notebook Section named “Load and explore the data.” Then, Click Data (the 0100 Icon) ⇨ Connections ⇨ Insert to code ⇨ Insert pandas DataFrame.



__40. In the pop-up to Select schema and table, Click ADMIN ⇨ CUSTOMER_CHURN ⇨ Select. This will insert the code needed to load the data table from NPS into the notebook.



__41. Near the end of the inserted code there are two lines of code that start with data_df_#, where the # is a number. If the number is not equal to 1, then change it to 1 in both lines of code.

data_df_4 ⇨ data_df_1

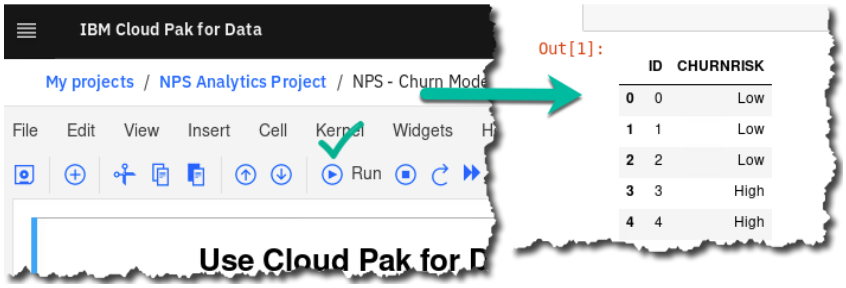
```
from project_lib import Project
project = Project.access()
NPS_Database_credentials = project.get_connection(name="NPS_Database")

import jaydebeapi, pandas as pd
NPS_Database_connection = jaydebeapi.connect('org.netezza.Driver',
    '{}://{}:{}/{}'.format('jdbc:netezza',
    NPS_Database_credentials['host'],
    NPS_Database_credentials['port'],
    NPS_Database_credentials['database']),
    [NPS_Database_credentials['username'],
    NPS_Database_credentials['password']])

query = 'SELECT * FROM ADMIN.CUSTOMER_CHURN'
data_df_1 = pd.read_sql(query, con=NPS_Database_connection)
data_df_1.head()

# You can close the database connection with the following code.
# NPS_Database_connection.close()
```

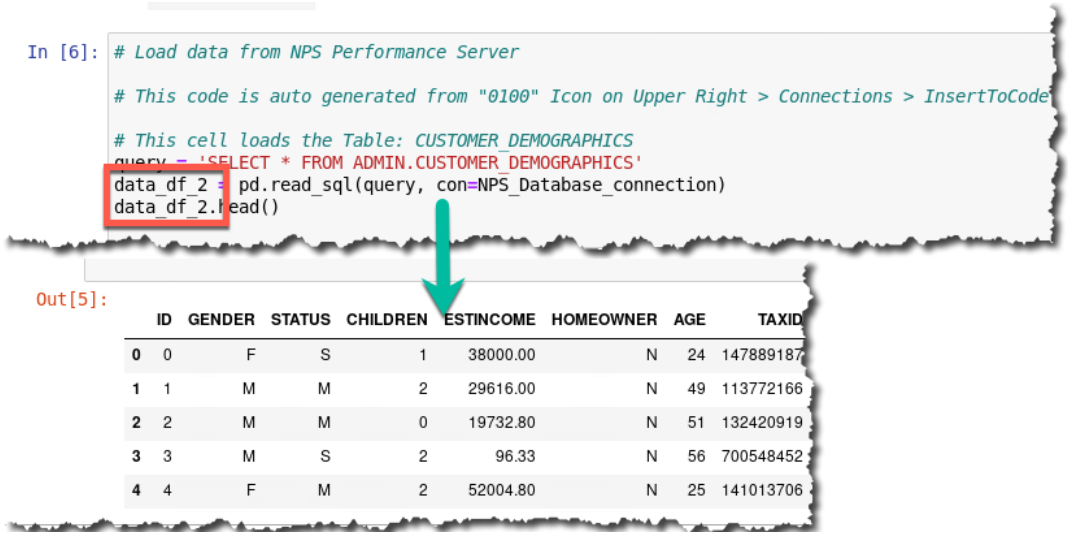
__42. Click Run to load the table into the notebook and display the first few rows.



	ID	CHURNRISK
0	0	Low
1	1	Low
2	2	Low
3	3	High
4	4	High

__43. Now you should be on the cell that loads the CUSTOMER_DEMOGRAPHICS table from NPS. To do so, repeat steps from the top of this sub-section Loading data into the Notebook.

Be sure to select the CUSTOMER_DEMOGRAPHICS table and use data_df_2 as the DataFrame variable name. Note that the code to load this table is shorter than the code to load the first table. This is because the code for the first table included loading the database credentials, which is only required once.



	ID	GENDER	STATUS	CHILDREN	ESTINCOME	HOMEOWNER	AGE	TAXID
0	0	F	S	1	38000.00	N	24	147889187
1	1	M	M	2	29616.00	N	49	113772166
2	2	M	M	0	19732.80	N	51	132420919
3	3	M	S	2	96.33	N	56	700548452
4	4	F	M	2	52004.80	N	25	141013706

 Data Scientist	Note: The code to load this table is shorter than the code to load the first table. This is because the code for the first table included loading the database credentials, which is only required once.

__44. Now you should be on the cell that loads the third and final table, CUSTOMER_ACTIVITY, from NPS. To do so repeat the steps again from sub-section `Loading data into the notebook`.

The only changes are to make sure to select the `CUSTOMER_ACTIVTY` table and use `data_df_3` as the DataFrame variable name.

```
In [7]: # Load data from NPS Performance Server

        # This code is auto generated from "0100" Icon on Upper Right > Connections > InsertToCode

        # This cell loads the Table: CUSTOMER_ACTIVITY
        query = 'SELECT * FROM ADMIN.CUSTOMER_ACTIVITY'
        data_df_3 = pd.read_sql(query, con=NPS_Database_connection)
        data_df_3.head()
```

	ID	TOTALDALLARVALUETRADED	TOTALUNITSTRADED	LARGESTSINGLETRANSACTION	S
0	593	1184043.01	488	9595.18	
1	702	1301994.91	520	9420.20	
2	1115	1207386.21	479	9395.26	
3	1780	1194102.87	521	9158.42	
4	1286	1272265.01	501	9518.61	

__45. Continue running the notebook one cell at a time until you reach the end. Near the end of the notebook a new deployment space will be created and the model that you've trained will be deployed as an online deployment. You'll work with that deployed model in the next Section.

```
                'id': '211e6bbf-4c2a-4b45-9e79-055d2a7ecd7e',
                'modified_at': '2020-07-29T16:56:27.636Z',
                'name': 'Churn-Model-Deployment',
                'parent': {'href': ''},
                'space_id': 'ccfeee5b-79c1-47ef-bcf1-eac8ba35048f',
                'tags': ['Churn-Model-Deployment-Tag']}}

In [50]: payload = [{"values": [ [1, 38000, 24, 1200000, 509, 9400, 940, 51, 3, 10, 0, -81000]]}]

In [51]: payload_metadata = {client.deployments.ScoringMetaNames.INPUT_DATA: payload}
         # score
         predictions = client.deployments.score(scoring_deployment_id, payload_metadata)
         predictions

Out[51]: {'predictions': [{'fields': ['prediction', 'probability'],
                           'values': [[0, [0.6454002261161804, 0.3545997738838196]]]}]}
```

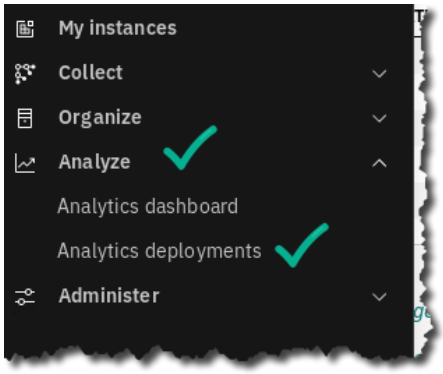
5. Conclusion

You have successfully completed this notebook!

15.8 Working with Batch Deployments

In this section we'll work with the model you saved from the Jupyter notebook.

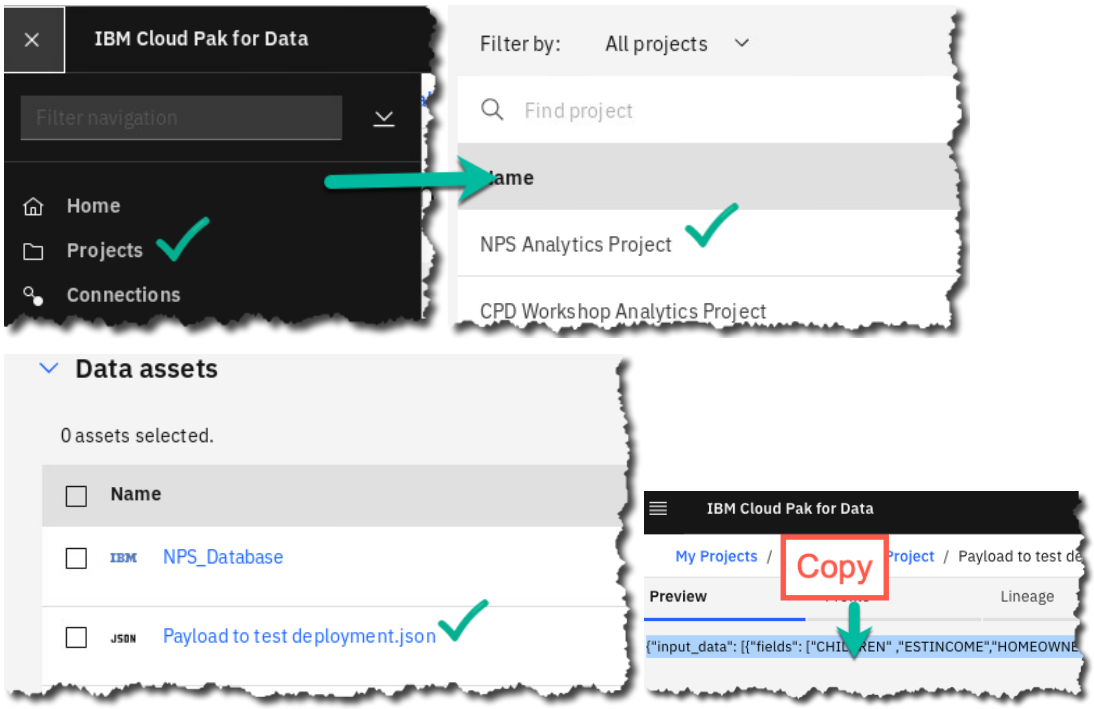
__46. Click `Navigation Menu` ⇨ `Analyze` ⇨ `Analytics deployments.`



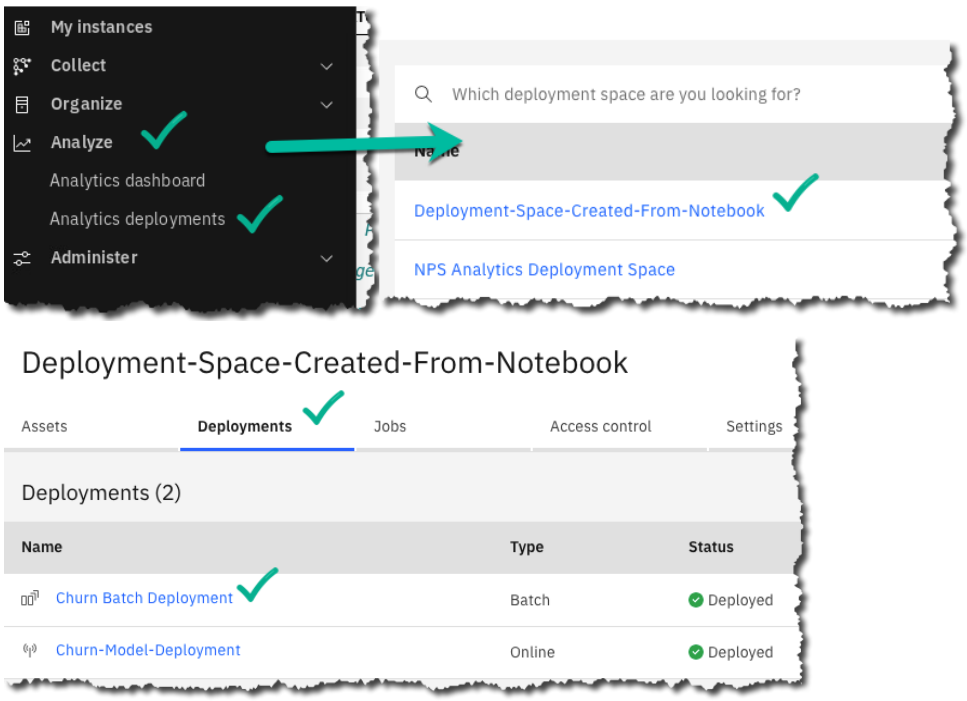
__47. Click `Deployment-Space-Created-From-Notebook` ⇨ `Churn-Model` ⇨ `Create deployment` ⇨ `Batch` ⇨ name it `Churn Batch Deployment` ⇨ select `One Standard CPU` ⇨ Click `Create`

__48. Before creating the batch job, we need to get some input data.

Click `Navigation Menu` ⇨ Click `Projects` ⇨ Click `NPS Analytics Project` ⇨ Under Data assets click `Payload to test deployment.json` ⇨ Carefully copy the entire contents of the json file (making sure to highlight everything).



__49. Click `Navigation Menu` ⇨ `Analyze` ⇨ `Analytics deployments` ⇨ `Deployment-Space-Created-From-Notebook` ⇨ `Deployments` ⇨ `Churn Batch Deployment.`



__50. Click `Create Job` ⇨ name the job `Churn Batch Job` ⇨ `Specify input data` ⇨ Click `Inline input data` ⇨ Paste the JSON data you copied in the previous step ⇨ `Confirm` ⇨ `Create and Run.`

__51. Let the job complete (takes about 20 seconds).

Click the `start time for the job run.`

Click `Show More` ⇨ scroll to the bottom of the log to view the prediction and probability.

15.9 Lab Conclusion

In this lab, you used tables stored in NPS as a source of training data for building churn models.

Using SPSS Modeler, part of the SPSS Modeler Premium Cartridge in Cloud Pak for Data, you built a churn model using the clickable paradigm for modeling. You deployed the model for online use and tested the deployment.

Using a Jupyter notebook written in Python you built another churn model using the coding paradigm for modeling. You deployed this model programmatically from the notebook and manually from the deployment space. You then tested the batch deployment using a JSON payload.

**** End of Lab 15 – SPSS using NPS – Deeper Dive**

Lab by Tom Konchan, Edited by Burt Vialpando, John Lucas and Kent Rubin

Lab 16 NETEZZA PERFORMANCE SERVER (NPS) GETTING STARTED

16.1 Lab overview

This lab will introduce you to the Netezza Performance Server (NPS) Data Warehouse service. We will also refer to NPS as Netezza. You will learn how to access `nzsql` from the Netezza command line tool and perform these basic operations:

- Create a database
- Create a table
- Load a table
- Unload a table

These tasks provide an initial experience of interacting with the database engine. Other labs (outside of this workshop) are available to provide users experience with the advanced features of Netezza.

16.2 Understanding Netezza Users

The NPS environment consists of two types of users - operating system users and database users. Operating system users can access the command line on the Netezza Linux host containers. Database users can access NPS to perform database-related tasks.

The Netezza Linux Container has the following default users:

Users	Capabilities
 root	Linux user with login permission to NPS Has ownership to various NPS files in the installation directory Usually not exposed to client users
 nz	Linux user with login permission to NPS Has ownership to various NPS files in the installation directory Access to the Netezza command tools
 admin	Database super-user (not a Linux user) Full access to all database administration privileges and object

16.3 Connecting to the Netezza Performance Server Command Line

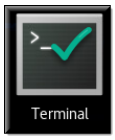
To access the Netezza host Linux container command line, you will use the Terminal application available on the Unified Desktop.

The Netezza host Linux container is running the service `sshd` which is configured to allow remote secure login. In our environment `ssh` is listening on port `22`.

nz	Note: In Cloud Pak for Data System with Netezza Performance Server, `sshd` is listening on port `51022`. In Netezza on Cloud `ssh` is disabled as the cloud solution provides other mechanisms to access the command line tools.

16.3.1 SSH to the Netezza Host Container using the Terminal

__1 Double-click the `Terminal` desktop icon.

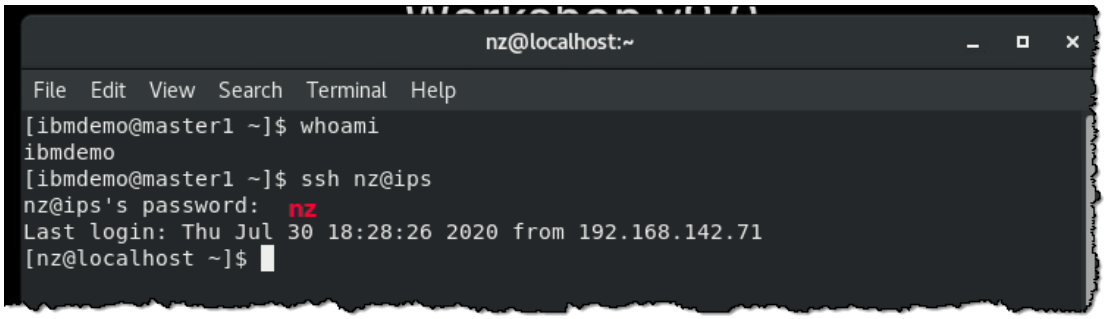


nz	Note: You can be user ibmdemo to ssh into node `ips`.

__2 Next, `ssh` into the Netezza Linux host container:

$ `whoami`

$ `ssh nz@ips` (password = `nz`)



__3 You are now logged into the Netezza host Linux container as user `nz`.

nz	Note: In this lab the root user is not needed on the Netezza host Linux container. For reference, the `root` password is `netezza`.

16.4 Checking the state of NPS

__4 Check the state of the Netezza engine by running `nzstate.`

$ `nzstate`

```
Last login: Thu Jul 30 18:31:42 2020 from 192.168.142.71
[nz@localhost ~]$ nzstate
System state is 'Online'.
[nz@localhost ~]$
```

If the Netezza system is offline (Stopped) then start Netezza with the `nzstart` command.

```
nz@localhost:~
File  Edit  View  Search  Terminal  Help
[nz@localhost ~]$ nzstate
System state is 'Stopped'.
[nz@localhost ~]$ nzstart
nzstart: Warning: Using auto-generated topology: /tmp/initTopology.autodb.sim.cfg

(startupsvr) Info: NZ-00022: --- program 'startupsvr' (25246) starting on host 'localhost.localdomain' ... ---
[nz@localhost ~]$ nzstate
System state is 'Online'.
[nz@localhost ~]$
```

16.5 Setting up the lab database

__5 Execute the script to set up the database used in this lab:

$ `cd ./labs/base/cli/setupLab/`

$ `ls -l`

$ `./setupLab.sh`

```
nz@localhost:~/labs/base/cli/setupLab
File  Edit  View  Search  Terminal  Help
[nz@localhost ~]$ cd ./labs/base/cli/setupLab
[nz@localhost setupLab]$ ls -l
total 8
-rwxr-xr-x. 1 nz nz 2607 May 11 07:47 labdb_tables.sql
-rwxr-xr-x. 1 nz nz 1045 May 11 07:47 setupLab.sh
[nz@localhost setupLab]$ ./setupLab.sh
ERROR:  DROP DATABASE: object LABDB does not exist.
CREATE DATABASE
ERROR:  DROP USER: object LABADMIN does not exist.
CREATE USER
ALTER USER
ALTER DATABASE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
CREATE TABLE
Load session of table 'NATION' completed successfully
Load session of table 'REGION' completed successfully
Load session of table 'CUSTOMER' completed successfully
Load session of table 'SUPPLIER' completed successfully
Load session of table 'PART' completed successfully
Load session of table 'PARTSUPP' completed successfully
Load session of table 'ORDERS' completed successfully
Load session of table 'LINEITEM' completed successfully
[nz@localhost setupLab]$
```

16.6 Connecting to the Netezza system database using nzsql

Since you have not created any user or databases yet, you will connect to the default database as the default user, with the following credentials:

Database: `system`
Username: `admin`
Password: `password`

When issuing the `nzsql` command, the user supplies the user account, password and the database to connect to, using the example syntax below. Do not try to execute this command as it is just demonstrating the syntax:

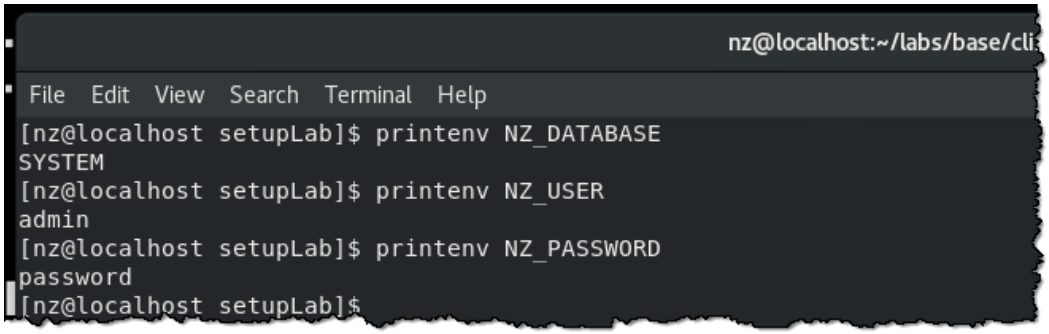
```
nzsql -d [db_name] -u [user] -pw [password]
```

Alternatively, these values (db_name, user, password) can be stored in the command shell and passed to the nzsql command when nzsql is issued without any arguments.

16.6.1 Verify the current database, user and password values

__6 Let's verify the current database, user and password values stored in the command shell by issuing `printenv NZ_DATABASE`, `printenv NZ_USER`, and `printenv NZ_PASSWORD` commands:

```
$ printenv NZ_DATABASE
$ printenv NZ_USER
$ printenv NZ_PASSWORD
```

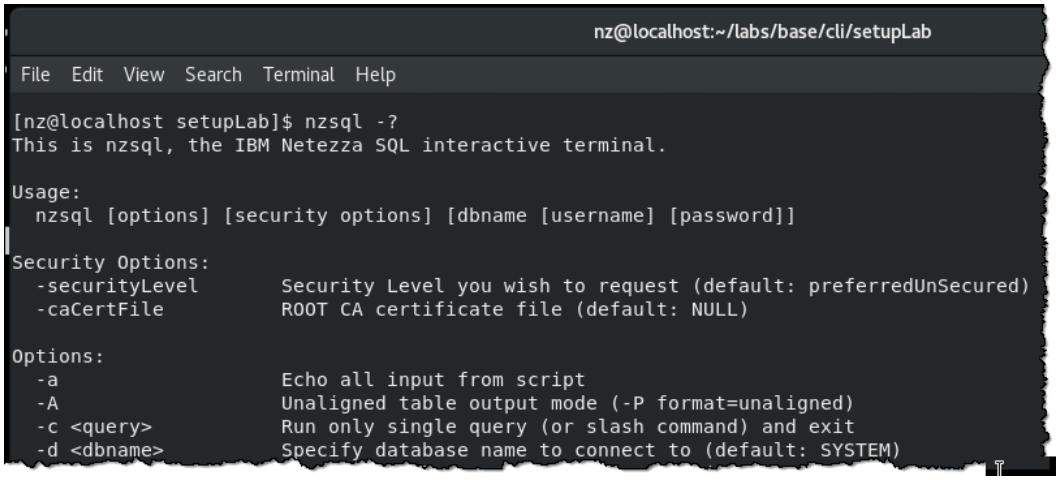


Since the current values correspond to our desired values, no modification is required. These environment variables are stored in the nz user's `.bashrc` file.

16.6.2 Review nzsql options and using the console

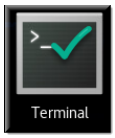
__7 Let's take a look at what options are available to start nzsql.

$ `nzsql -?`



__8 The `-?` option will list the usage and all options for the `nzsql` command. In this next exercise, you will start `nzsql` without arguments.

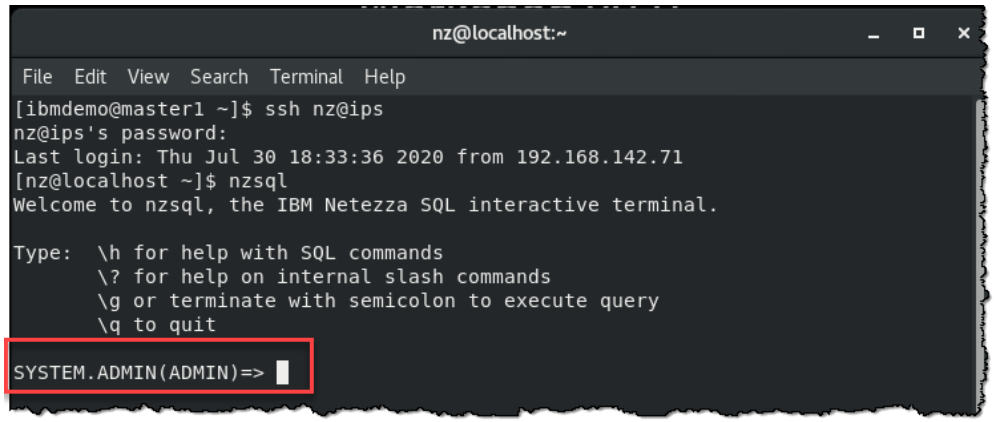
Open up another terminal window by double-clicking the `Terminal` desktop icon.



__9 Next, `ssh` into the Netezza Linux host container:

$ `ssh nz@ips` (password = `nz`)

$ `nzsql`



This will bring up the `nzsql` prompt below that shows a connection to the system database as user admin:

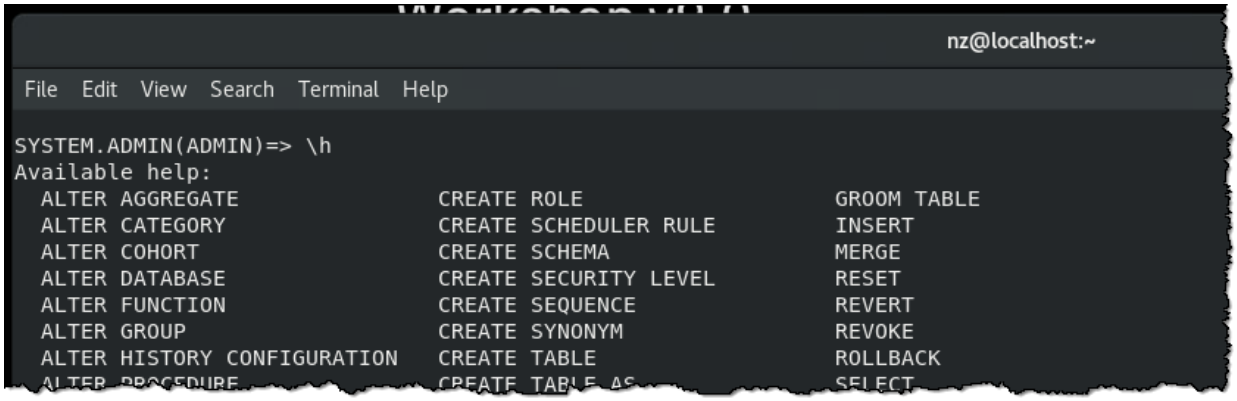
The `nzsql` console prompt format: `DATABASE.SCHEMA(USER)`

16.7 Commonly used commands and SQL statements

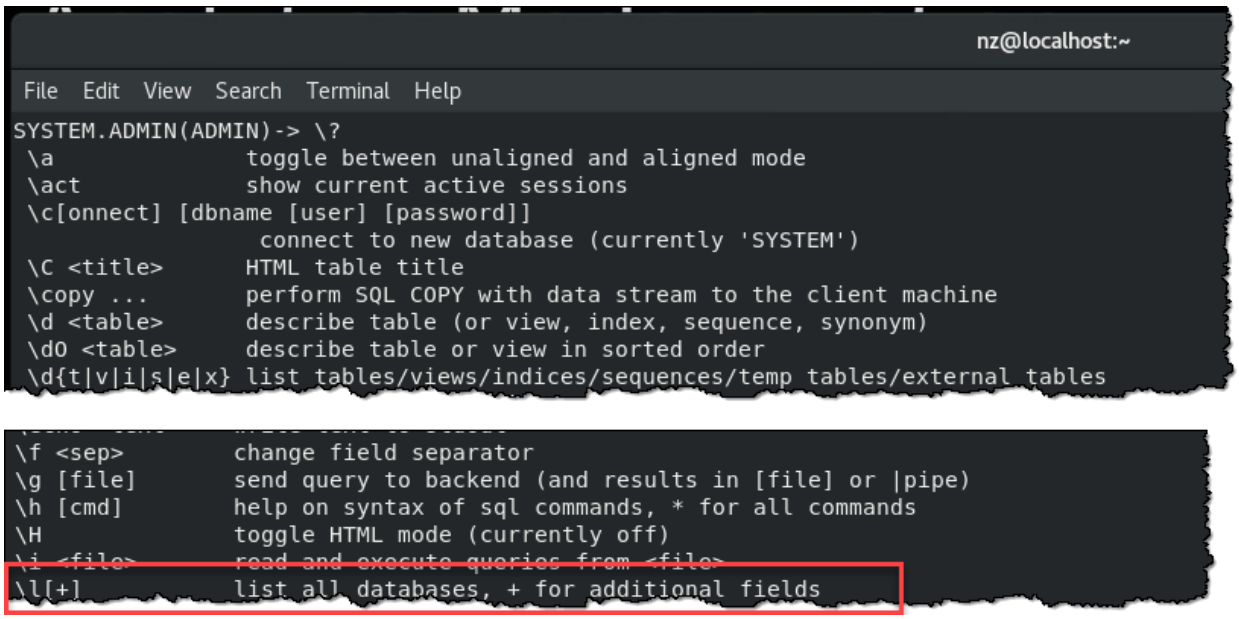
There are commonly used commands that start with \ which we will demonstrate in this section. First, you will run the two help commands to familiarize yourself with these handy commands. The \h command lists the available SQL commands, while the \? command is used to list the internal slash commands.

__10 Execute these commands in the nzsql console:

```
SYSTEM.ADMIN(ADMIN)=> \h
```



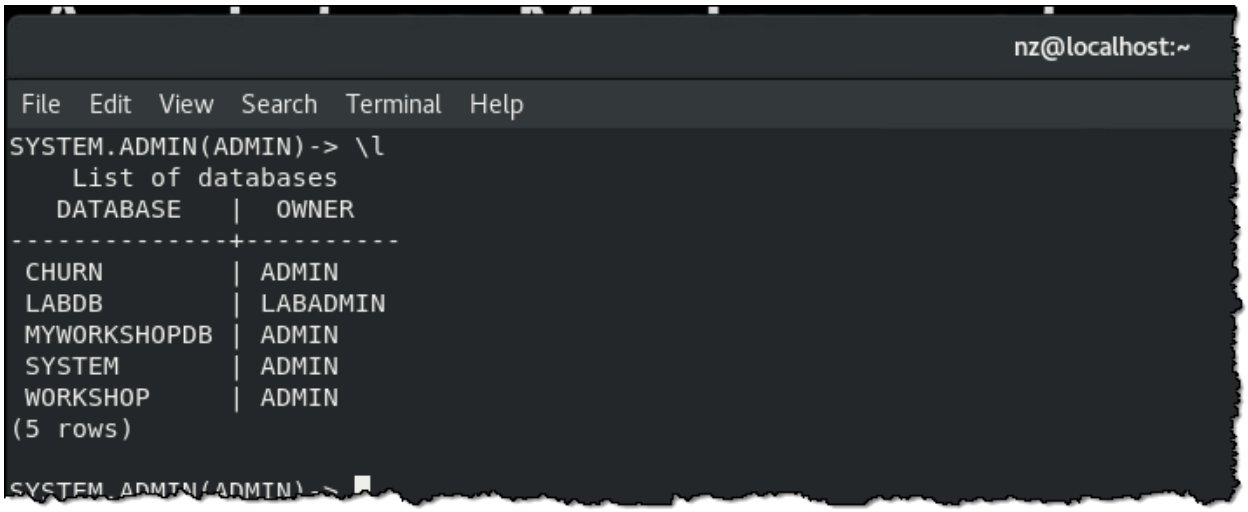

```
SYSTEM.ADMIN(ADMIN)=> \?
```



From the output of the \? command, you found the \l internal command you can use to find out all the databases.

__11 Let's look at the list of databases that have been created:

```
SYSTEM.ADMIN(ADMIN)=> \l
```

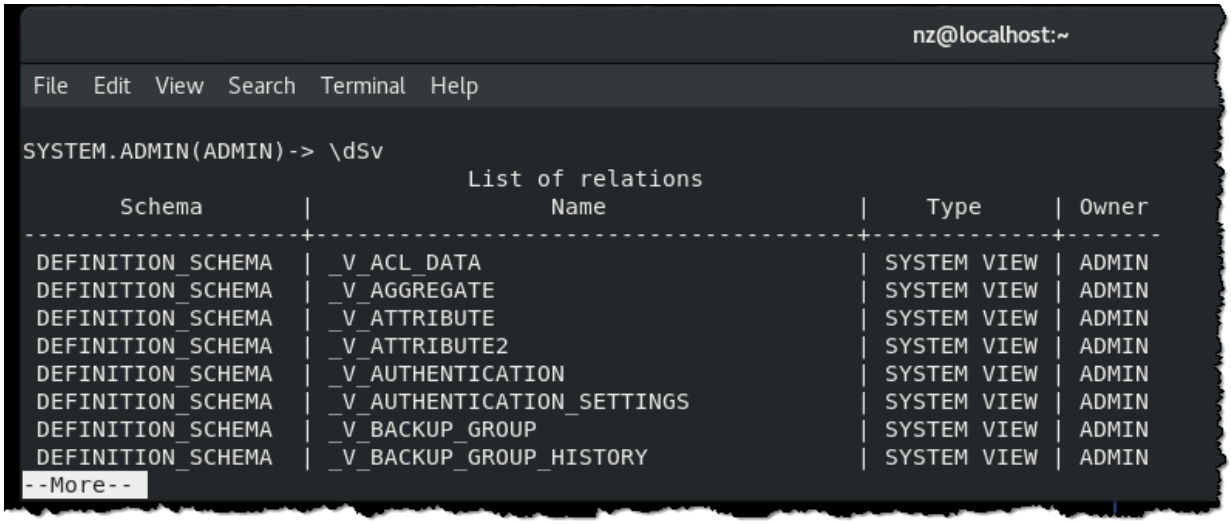


Secondly, you will use `\dSv` to find the system views within the system database.

nz	Note: There are system tables, but it is not recommended to directly access those tables as they can change from release to release and are restricted from the normal user.

__12 Let's find the system views:

```
SYSTEM.ADMIN(ADMIN)=> \dSv
```



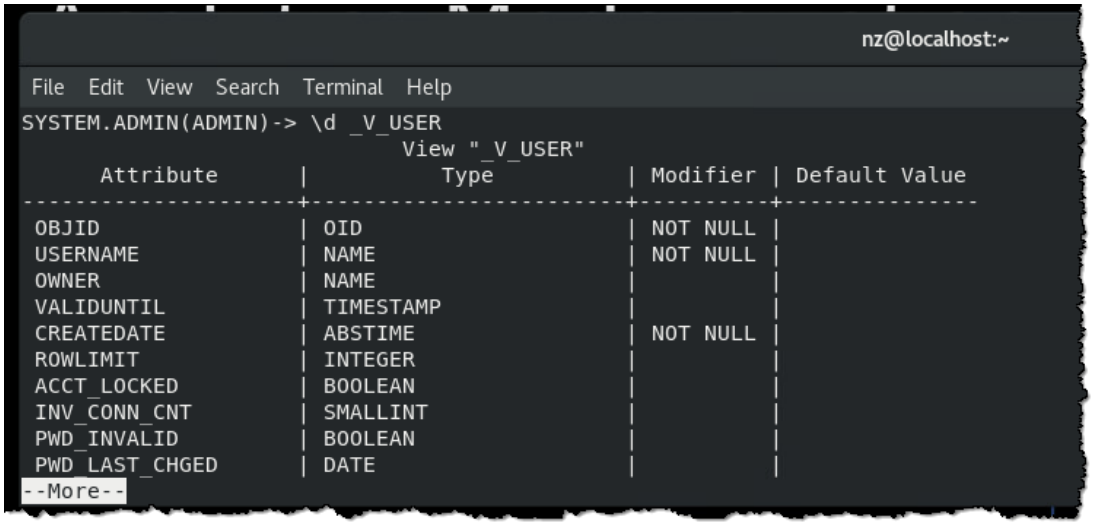
nz	Note: Press the space bar to scroll down the result set when you see `--More--` on the screen.

The list of the system views was truncated in the output above due to the length. Here are the primary views to investigate for DBAs/users new to Netezza:

```
_V_USER
_V_SESSION_V_DATABASE
_V_SCHEMA
_V_TABLE
_V_VIEW
_V_ATTRIBUTE
_V_SEQUENCE
_V_SYNONYM
_V_FUNCTION
_V_AGGREGATE
_V_PROCEDURE
```

__13 From the previous command, you can see that there is a user table called _V_USER. To find out what is stored in that table, you will use the describe command \d.

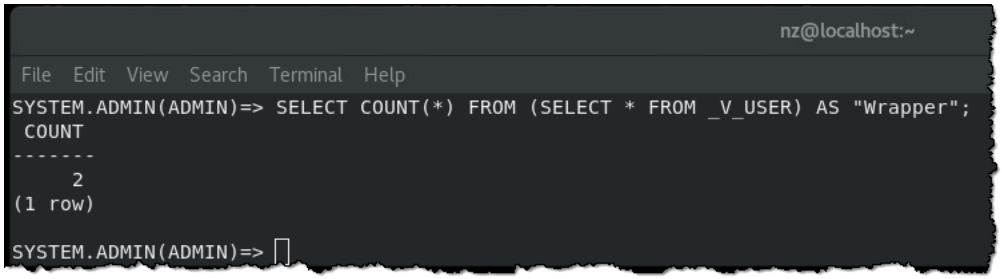
SYSTEM.ADMIN(ADMIN)=> \d _V_USER



This will return all the columns of the _V_USER system table.

__14 Next, find the existing users stored in the table. In case too many rows are returned at once, you can first calculate the number of rows it contains by executing the following query:

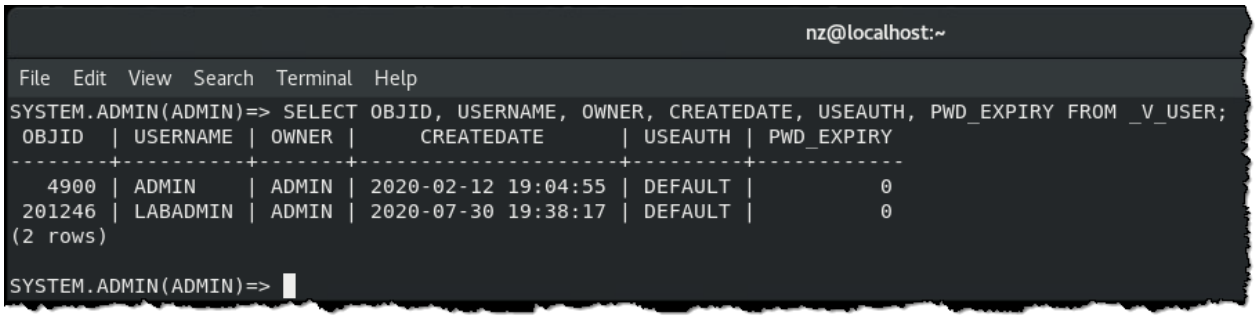
SYSTEM.ADMIN(ADMIN)=> SELECT COUNT(*) FROM (SELECT * FROM _V_USER) AS "Wrapper";



__15 The above query is essentially the same as `SELECT COUNT (*) FROM _V_USER.` We have demonstrated the subselect syntax in case there is a complex query that needs to have the result set evaluated. The result should show the two entries in the user table.

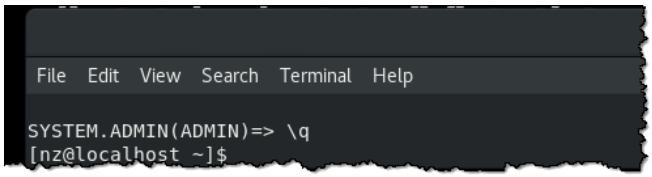
You can next enter the following query to list the usernames:

```
SYSTEM.ADMIN(ADMIN)=> SELECT OBJID, USERNAME, OWNER, CREATEDATE, USEAUTH,
PWD_EXPIRY FROM _V_USER;
```



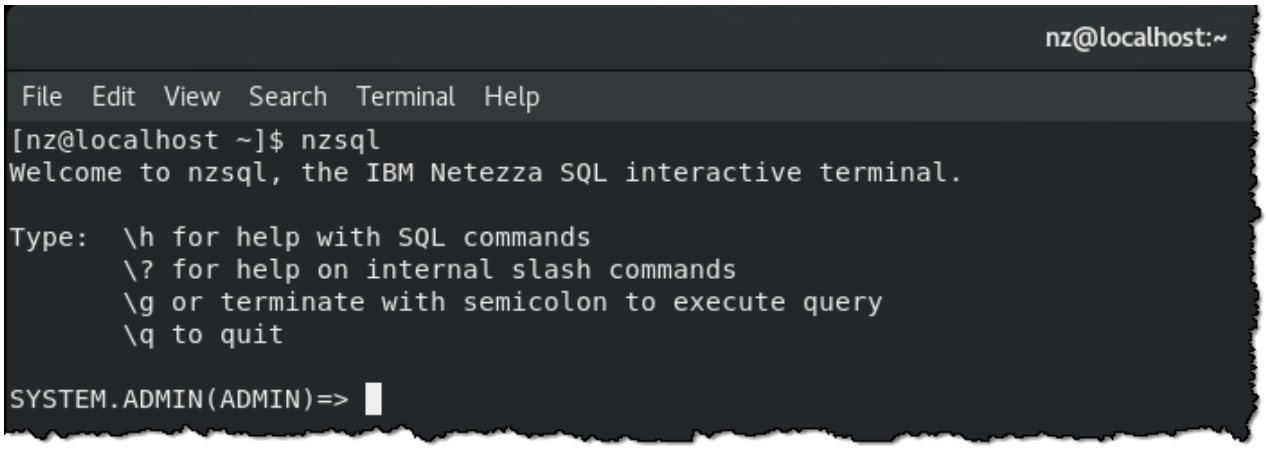
__16 To exit `nzsql`, use the command `\q` to return to NPS.

```
SYSTEM.ADMIN(ADMIN)=> \q
```



__17 Re-enter `nzsql:`

```
$ nzsql
```



You will be using the `nzsql` command line tool throughout the rest of this lab.

For a complete list of the nzsql internal slash options, see: https://ibm.biz/nzsql-slash-options.

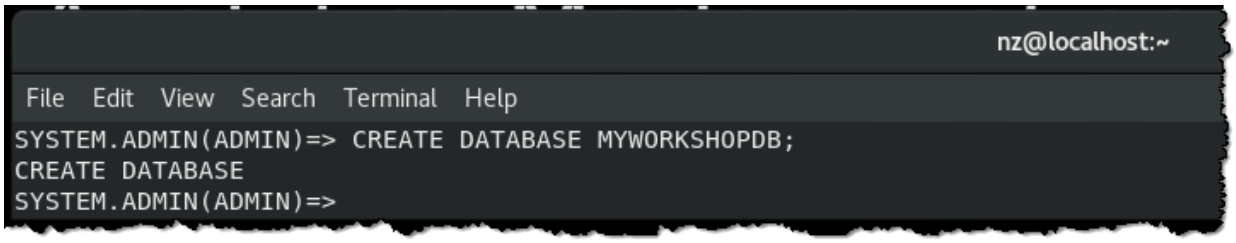
16.8 Creating a database

A database in Netezza is a collection of entities such as schemas, tables, views, synonyms, and other objects.

The initial system consists of the SYSTEM database containing system tables and views. Initially, only the admin user can create databases, but the admin user can grant other users permission to create databases as described in security and access control. You cannot delete the system database. The admin user can also make another user the owner of a database, giving that user admin-like control over that database and its contents. Security is out of scope for this lab as you will perform all operations as the ADMIN database super-user.

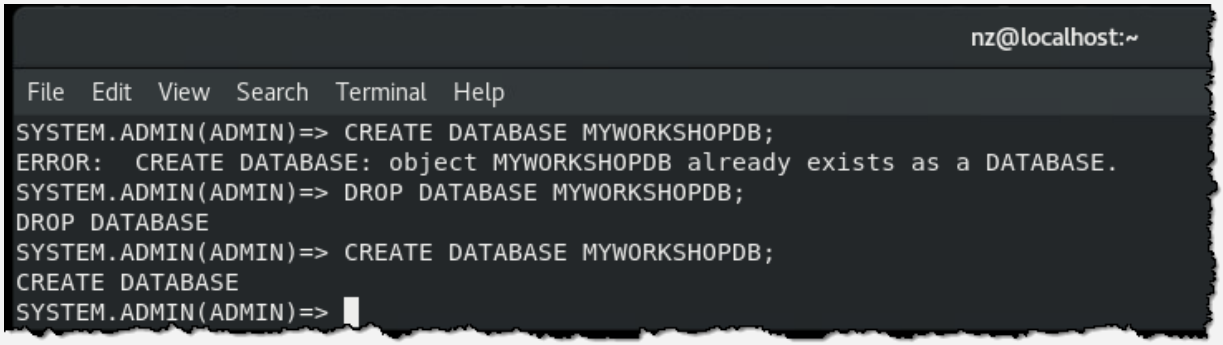
__18 The database creator becomes the default owner of the database. The owner can remove the database and all of its objects, even if other users own objects within the database.

```
SYSTEM.ADMIN(ADMIN)=> CREATE DATABASE MYWORKSHOPDB;
```



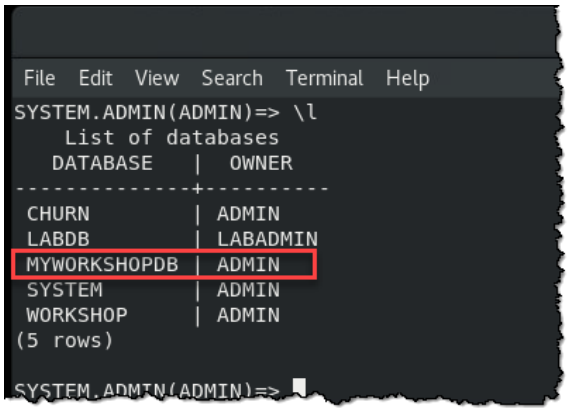
If database MYWORKSHOPDB already exists, you will get an error on the CREATE statement.

Simply drop it and then create it like this:



__19 List the databases in the Netezza system:

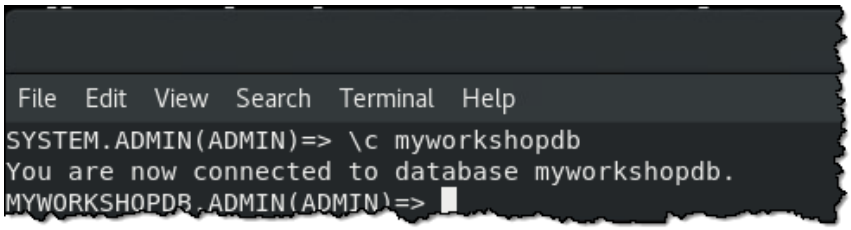
```
SYSTEM.ADMIN(ADMIN)=> \l
```



Notice the end of the `\` commands do not require a terminator `;` this is because the `\` commands terminate automatically on a single line. SQL statements can span more than one line in `nzsql`.

__20 Connect to the new database:

```
SYSTEM.ADMIN(ADMIN)=> \c myworkshopdb
```



16.9 Creating a table

Now that you have a database, you can create a table to store your data. Tables are specific to a Netezza database and have one or more columns, each with attributes or data types.

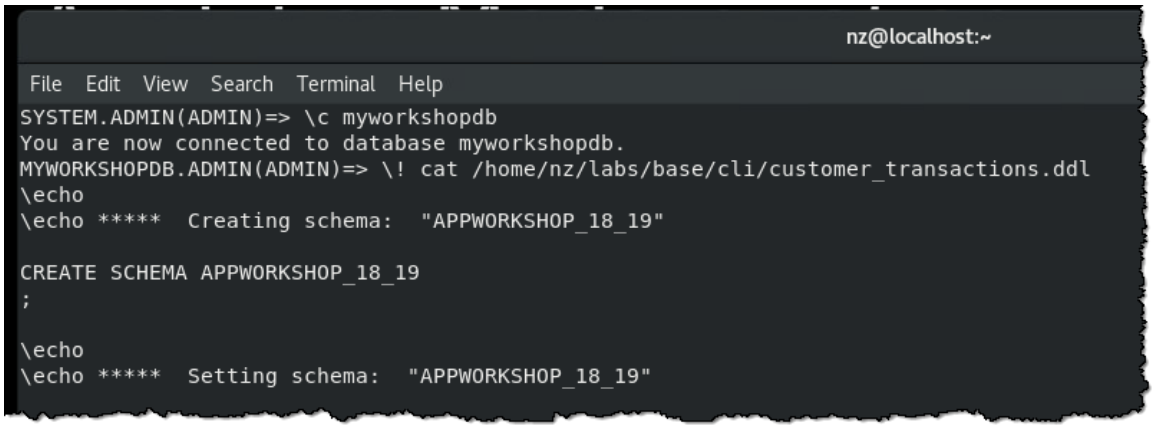
Netezza supported data types:

Data type	Example	Description
byteint smallint integer bigint	120 0 256 1290985	See Integer data types.
numeric decimal	-99.56 123.679	See Fixed-point data types.
real double precision	-81293.35	See Floating-point data types.
char (*n*)	salary	See Character strings.
varchar (*n*)	this is a variable string	See Character strings.
boolean	true	This is an ASCII string that contains one of the following values: True yes 1 t y False no 0 f n
JSON		See Document data types
date	2002-02-04	The date is an exact 4-byte data type. The system recognizes a range of dates that are composed of year, month, and day. See DateStyle option.
time	01:59:45 23:00:01	See time data type.
time with time zone	01:15:33 -05	See Time with time zone (timetz) data type.
timestamp	2002-02-04 01:15:33	See timestamp data type.
interval	2 years 3 months 4 days	See Interval data types.

__21 To create our lab table, make sure you are connected to the database you created called MYWORKSHOPDB. A DDL file was prepared for you in `/home/nz/labs/base/cli.`

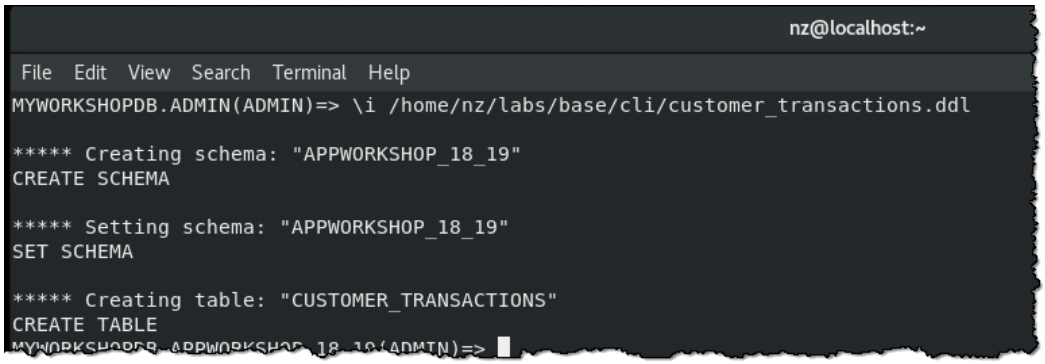
View the file while in nzsql by running an operating system command with `\!` option:

```
MYWORKSHOPDB.ADMIN(ADMIN)=> \! cat
/home/nz/labs/base/cli/customer_transactions.ddl
```



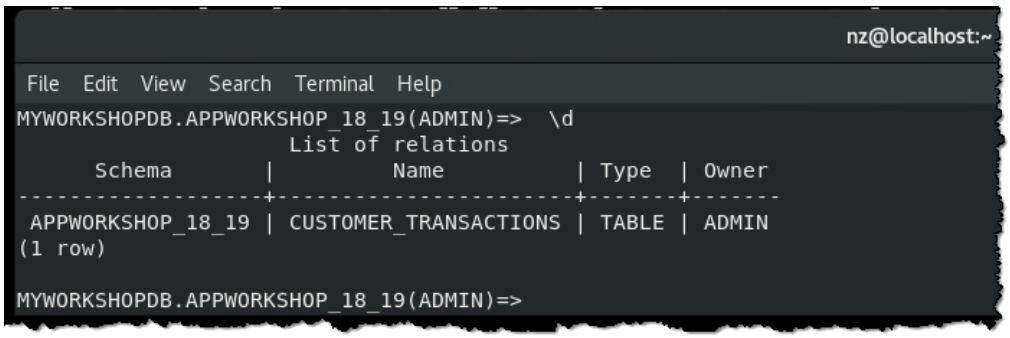
__22 Now create the table with the `\i` option:

```
MYWORKSHOPDB.ADMIN(ADMIN)=> \i
/home/nz/labs/base/cli/customer_transactions.ddl
```



__23 Check that the table exists: (Note: there is a space before the `\d`):

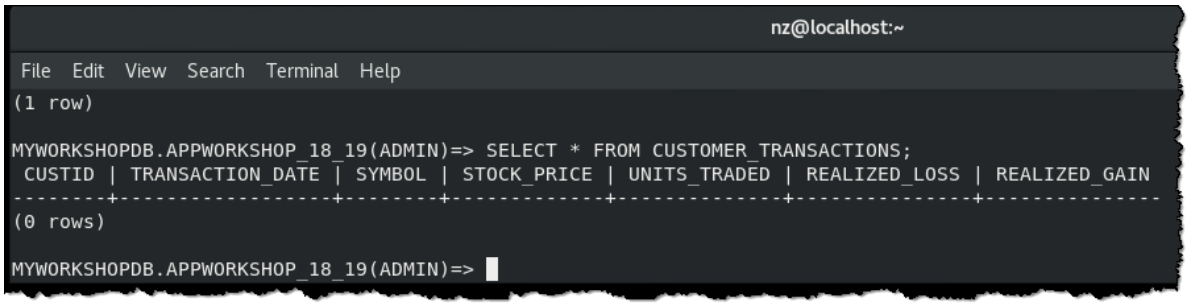
```
MYWORKSHOPDB.ADMIN(ADMIN)=>  \d
```



Notice that the `nzsql` prompt schema changed, this was due to the `SET SCHEMA` command that was specified in `customer_transactions.ddl`.

__24 Select the table data:

```
MYWORKSHOPDB.APPWORKSHOP_18_19(ADMIN)=> SELECT * FROM CUSTOMER_TRANSACTIONS;
```



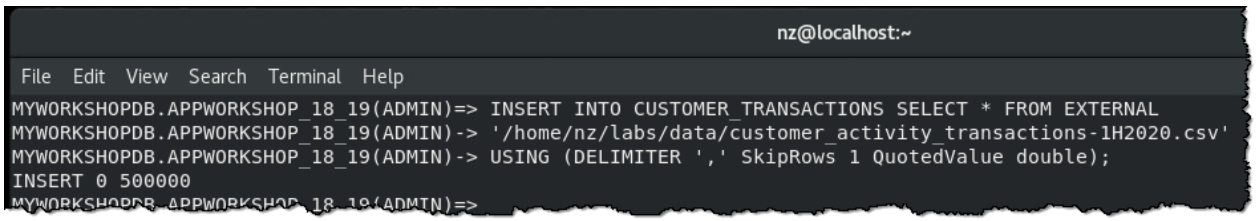
There is no data in the table yet because you just created it. In the next section, you will load some data.

16.10 Loading data into a table

__25 Load data into our CUSTOMER_TRANSACTIONS table using an external table (a csv file.).

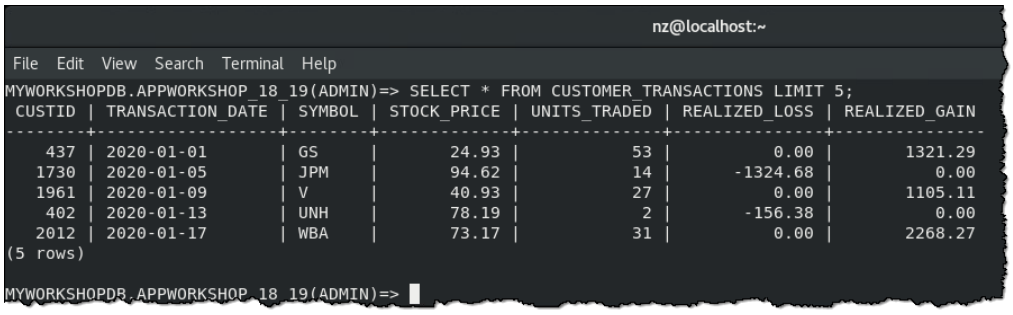
You will be using a three line command for doing this, terminated on the last line with the ; terminator:

```
MYWORKSHOPDB.APPWORKSHOP_18_19(ADMIN)=> INSERT INTO CUSTOMER_TRANSACTIONS
SELECT * FROM EXTERNAL
MYWORKSHOPDB.APPWORKSHOP_18_19(ADMIN)=>
'/home/nz/labs/data/customer_activity_transactions-1H2020.csv'
MYWORKSHOPDB.APPWORKSHOP_18_19(ADMIN)=> USING (DELIMITER ',' SkipRows 1
QuotedValue double);
```



__26 Select 5 rows from the table CUSTOMER_TRANSACTIONS:

```
MYWORKSHOPDB.APPWORKSHOP_18_19(ADMIN)=> SELECT * FROM CUSTOMER_TRANSACTIONS
LIMIT 5;
```



16.11 Lab conclusion

In this lab you learned how to use the `nzsql` tool to create a database, a table and load data into that table. These are the basic functions of the Netezza database.

** End of Lab 16 – Netezza Performance Server (NPS) Getting Started

Lab by Dan Hancock, Edited by Burt Vialpando and Kent Rubin

Back Page: Notices

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
USA

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice and represent goals and objectives only.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental. All references to fictitious companies or individuals are used for illustration purposes only.

Back Page: Trademarks and Copyrights

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	AIX	CICS	ClearCase	ClearQuest	Cloudscape
Cube Views	Db2	developerWorks	DRDA	IMS	IMS/ESA
Informix	Lotus	Lotus Workflow	MQSeries	OmniFind	
Rational	Redbooks	Red Brick	RequisitePro	System i	
System z	Tivoli	WebSphere	Workplace	System p	

Adobe, Acrobat, Portable Document Format (PDF), and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. See Java Guidelines

Microsoft, Windows, Windows NT, and the Windows logo are registered trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

ITIL is a registered trademark and a registered community trademark of the Office of Government Commerce and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Other company, product and service names may be trademarks or service marks of others.

NOTES

NOTES

Please Recycle

IBM Software