BOOTCAMP PROJECT 4

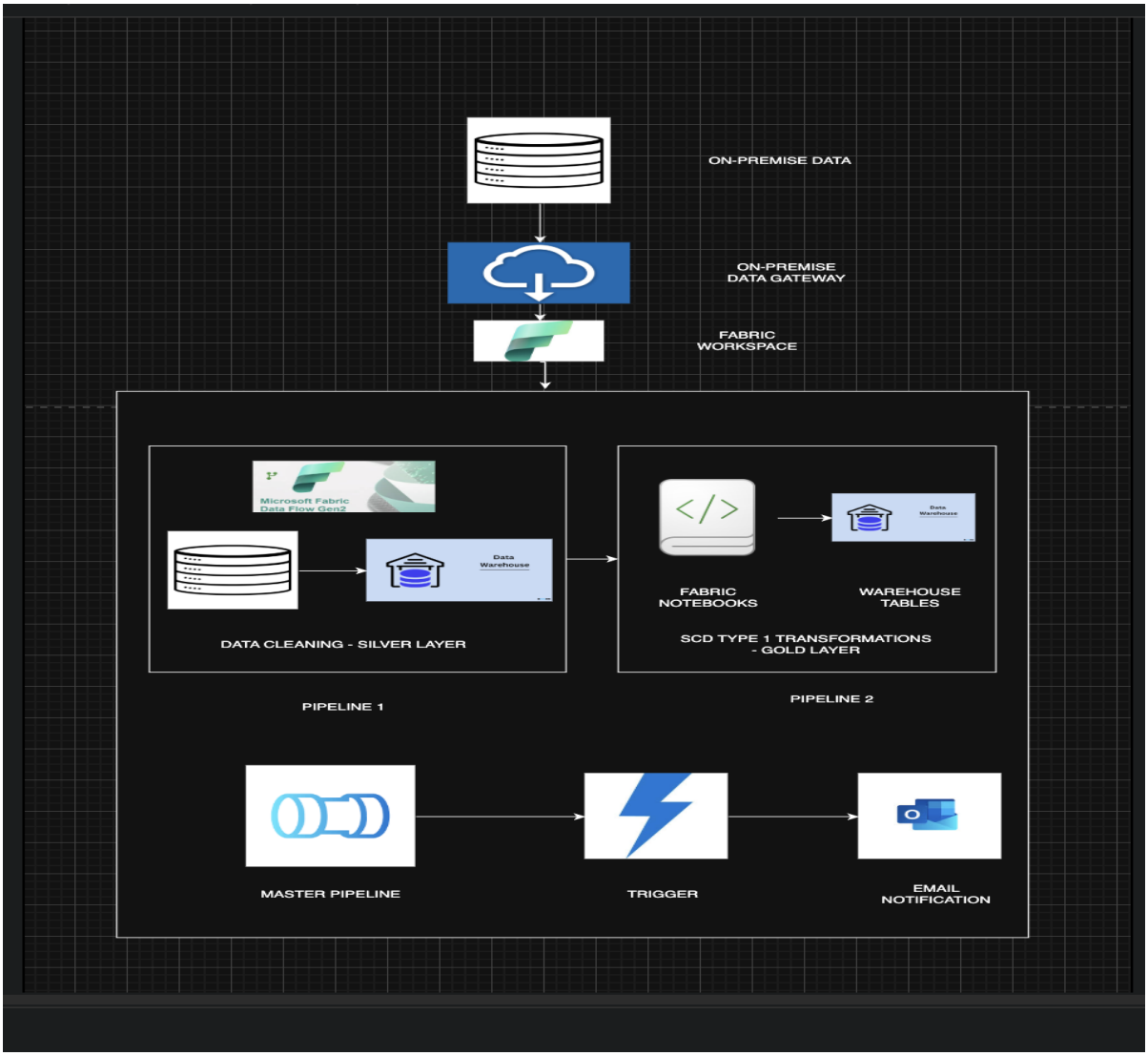
INCREMENTAL DATA LOADING AND AUTOMATED NOTIFICATIONS USING MICROSOFT FABRIC

SUBMITTED BY DRUSYA SURESH

OBJECTIVE:

To build an end-to-end data pipeline on Microsoft Fabric that:
1. Ingests structured data from on-premises environments into a Fabric Lakehouse using the On-Prem Gateway.
2. Utilizes the AI Bank Dataset as the source.
3. Implements Dataflow Gen 1 to join tables, remove duplicates, and clean data
Loads the cleansed data into a Fabric Warehouse
4.Applies Slowly Changing Dimension (SCD) Type 1 logic using Fabric Notebooks and writes the results into separate warehouse tables
5.Schedules and monitors the pipeline, sending an automated email notification (via Outlook or Gmail) upon successful pipeline completion.

ARCHITECTURE DIAGRAM:



STEPS INVOLVED:

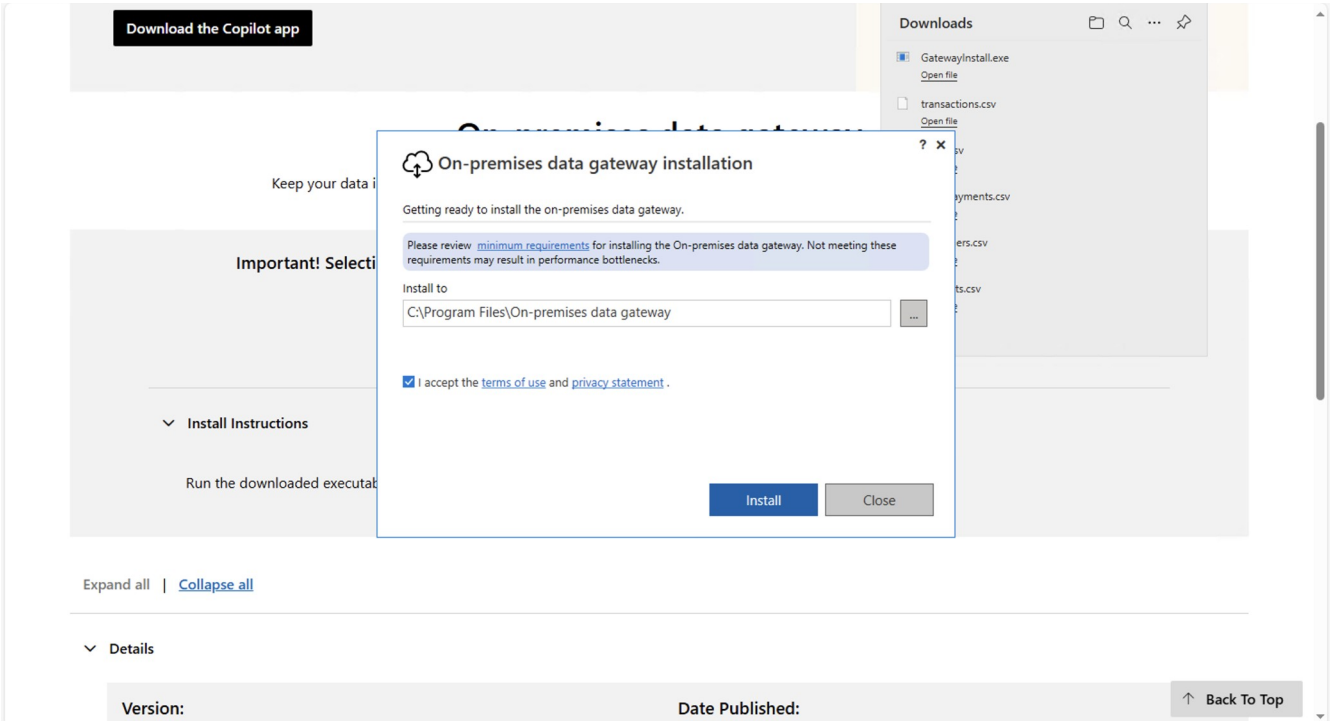
- Connect Microsoft Fabric with on-premise environment using the virtual machine and on-prem gateway.
- Copy the data in the on-prem to the data lake using pipeline activity and load them into lake house tables – Bronze layer.
- Using data flow activity clean and standardize the data and load them to data warehouse – Silver layer.
- Perform SCD Type 1 logic on the cleaned data using Fabric notebooks and load them into separate tables in warehouse – Gold layer.
- Using a master pipeline trigger all the activities so that can schedule to run it on specific time and also set an email notification upon successful pipeline completion.

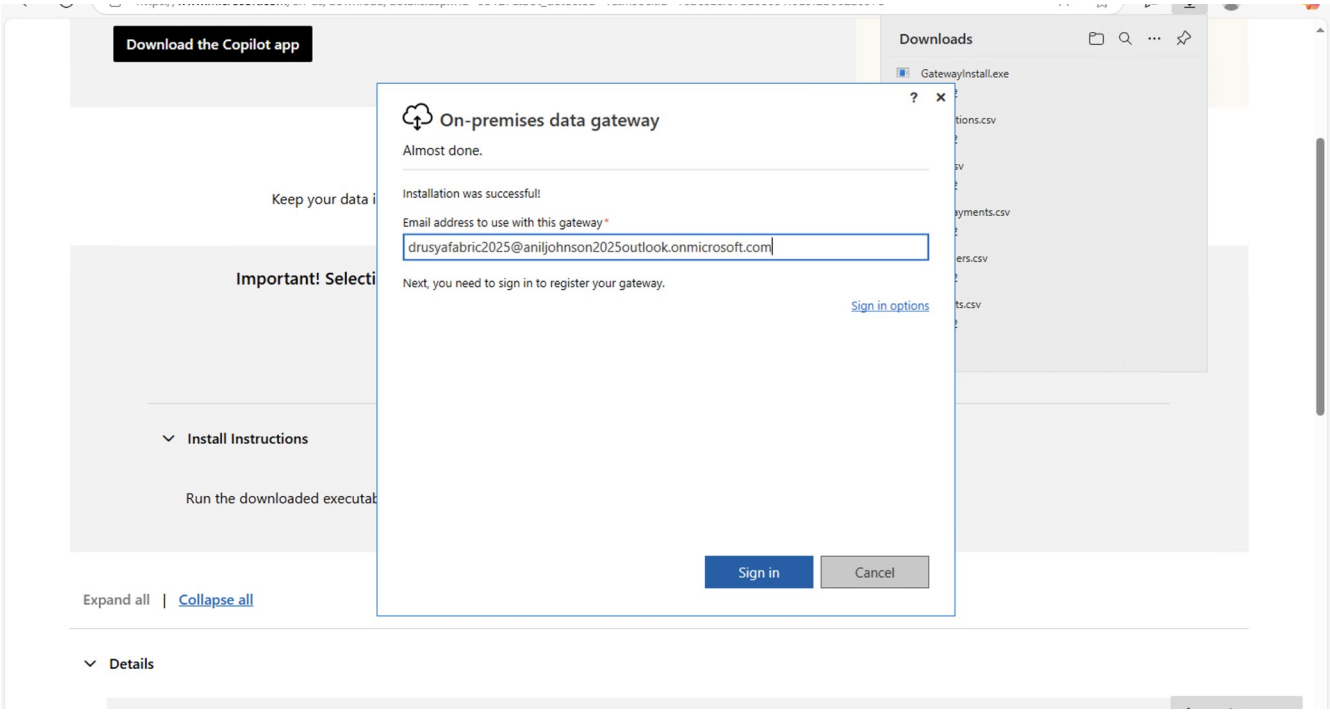
TOOLS AND TECHNOLOGIES USED:

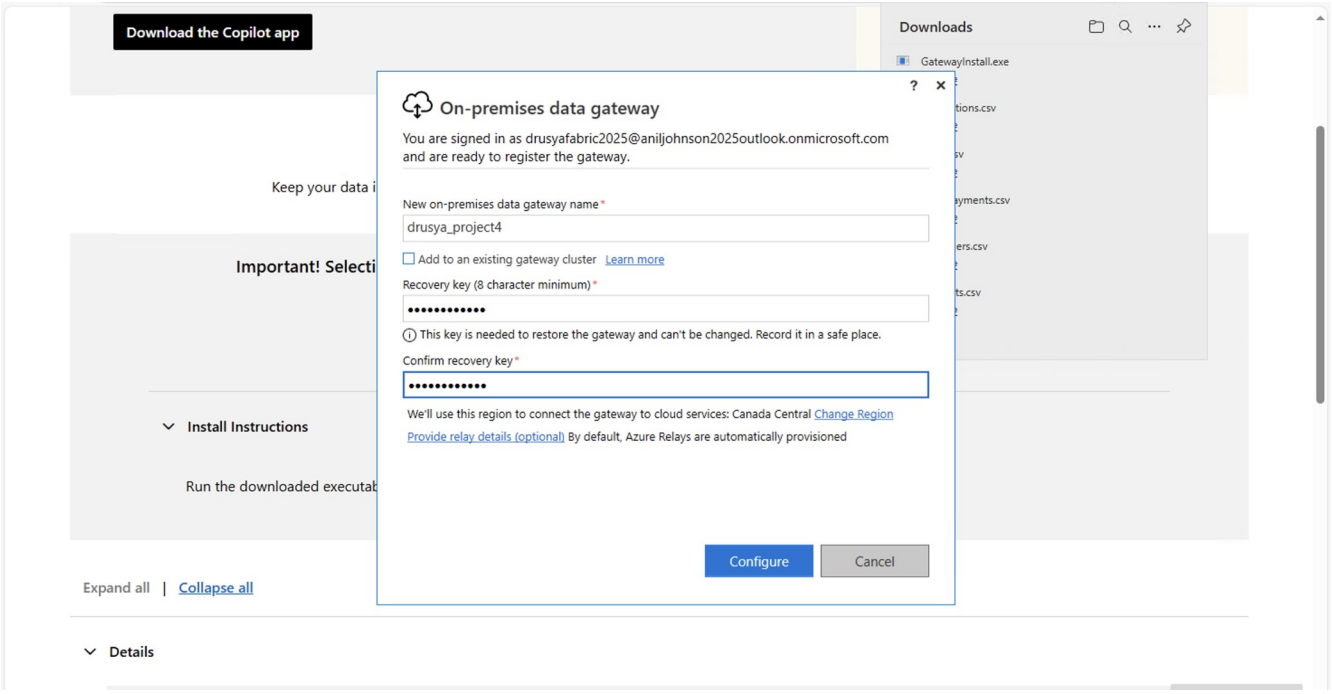
• Microsoft Fabric
• On-Premises Data Gateway
• Fabric Lakehouse and Warehouse
• Fabric Dataflow Gen 1
• Fabric Notebook
• Email Notification Task (in-built.

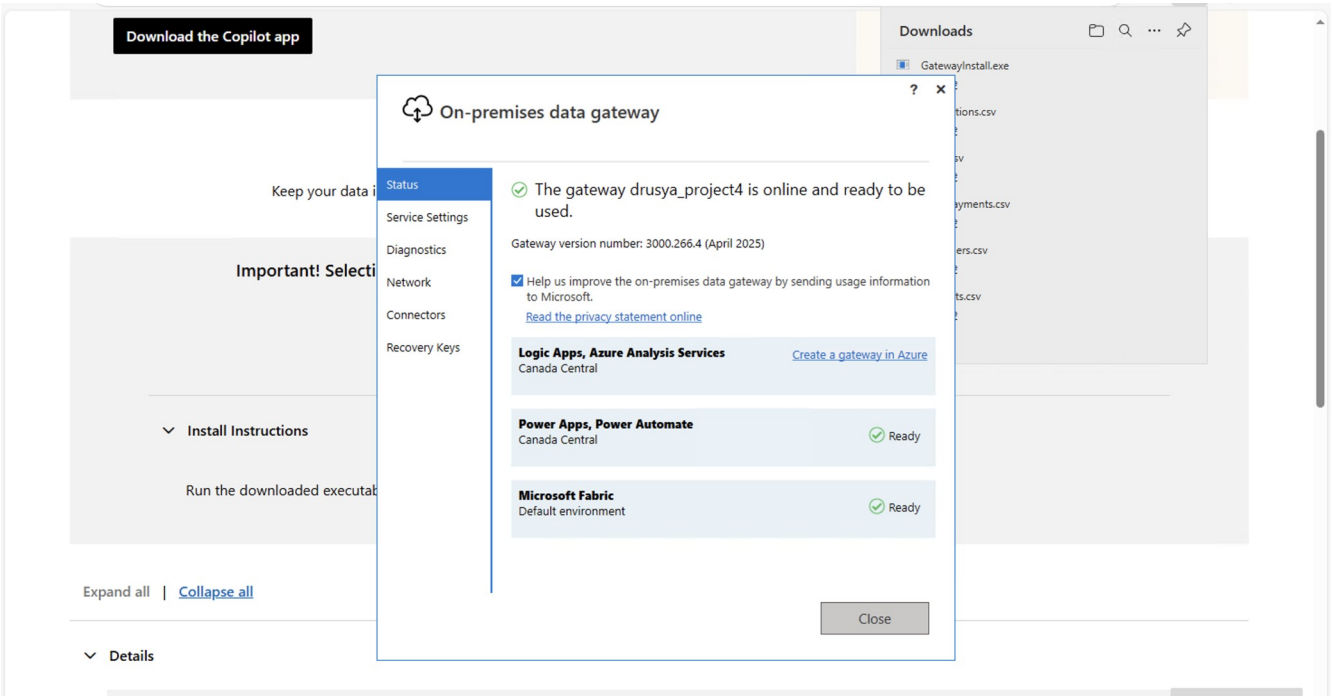
CONNECTING ON-PREMISE WITH FABRIC WORKSPACE USING ON-PREM DATA GATEWAY

For this I am using virtual machine to download the on-prem data gateway. I have my data in the virtual machine downloads folder.


Download the Copilot app
Downloads
GatewayInstall.exe
Open file
transactions.csv
Open file
On-premises data gateway installation
Getting ready to install the on-premises data gateway.
Please review minimum requirements for installing the On-premises data gateway. Not meeting these requirements may result in performance bottlenecks.
Install to
C:\Program Files\On-premises data gateway
I accept the terms of use and privacy statement .
Install
Close
Keep your data i
Important! Selecti
Install Instructions
Run the downloaded executab
Expand all | Collapse all
Details
Version:
Date Published:
Back To Top

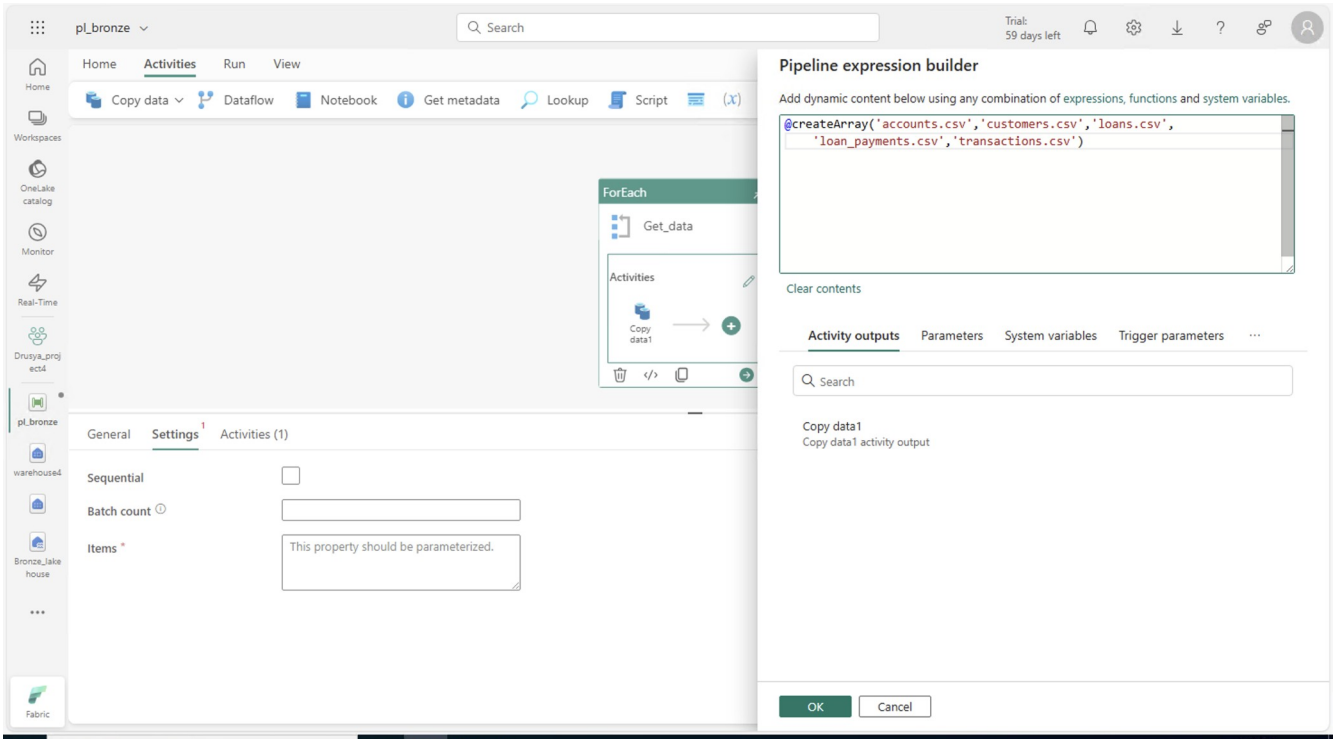

Download the Copilot app
Downloads
GatewayInstall.exe
On-premises data gateway
Almost done.
Installation was successful!
Email address to use with this gateway *
drusyafabric2025@aniljohnson2025outlook.onmicrosoft.com
Next, you need to sign in to register your gateway.
Sign in options
Sign in
Cancel
Keep your data i
Important! Selecti
Install Instructions
Run the downloaded executab
Expand all | Collapse all
Details


Download the Copilot app
Downloads
GatewayInstall.exe
On-premises data gateway
You are signed in as drusyafabric2025@aniljohnson2025outlook.onmicrosoft.com and are ready to register the gateway.
New on-premises data gateway name*
drusya_project4
Add to an existing gateway cluster Learn more
Recovery key (8 character minimum)*
This key is needed to restore the gateway and can't be changed. Record it in a safe place.
Confirm recovery key*
We'll use this region to connect the gateway to cloud services: Canada Central Change Region
Provide relay details (optional) By default, Azure Relays are automatically provisioned
Configure
Cancel
Keep your data i
Important! Selecti
Install Instructions
Run the downloaded executab
Expand all | Collapse all
Details


Download the Copilot app
Downloads
GatewayInstall.exe
On-premises data gateway
Status
Service Settings
Diagnostics
Network
Connectors
Recovery Keys
The gateway drusya_project4 is online and ready to be used.
Gateway version number: 3000.266.4 (April 2025)
Help us improve the on-premises data gateway by sending usage information to Microsoft.
Read the privacy statement online
Logic Apps, Azure Analysis Services
Canada Central
Create a gateway in Azure
Power Apps, Power Automate
Canada Central
Ready
Microsoft Fabric
Default environment
Ready
Close
Keep your data i
Important! Selecti
Install Instructions
Run the downloaded executab
Expand all | Collapse all
Details

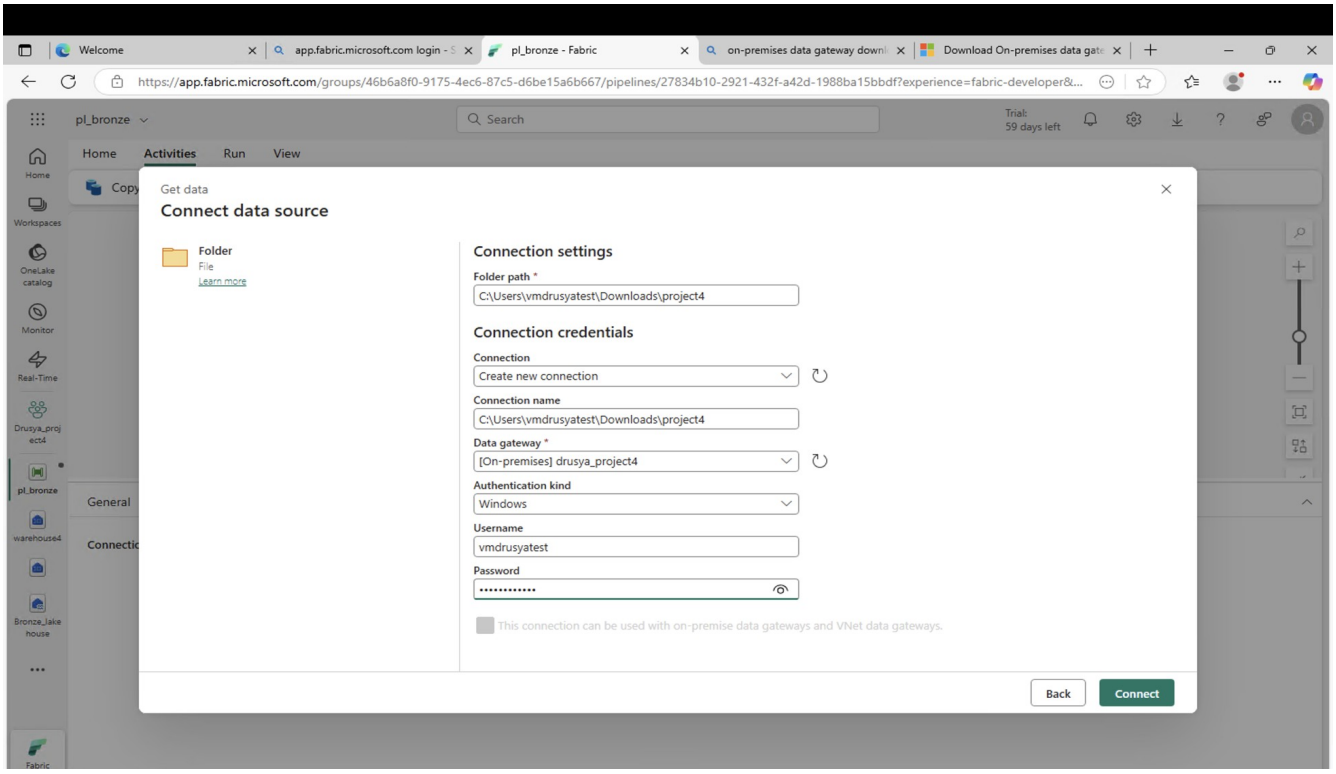
COPY THE FILES USING FOR EACH FROM ON-PREM TO LAKEHOUSE AS TABLES – BRONZE LAYER

For this we use a pipeline inside which we add a for each activity to pass all the files.

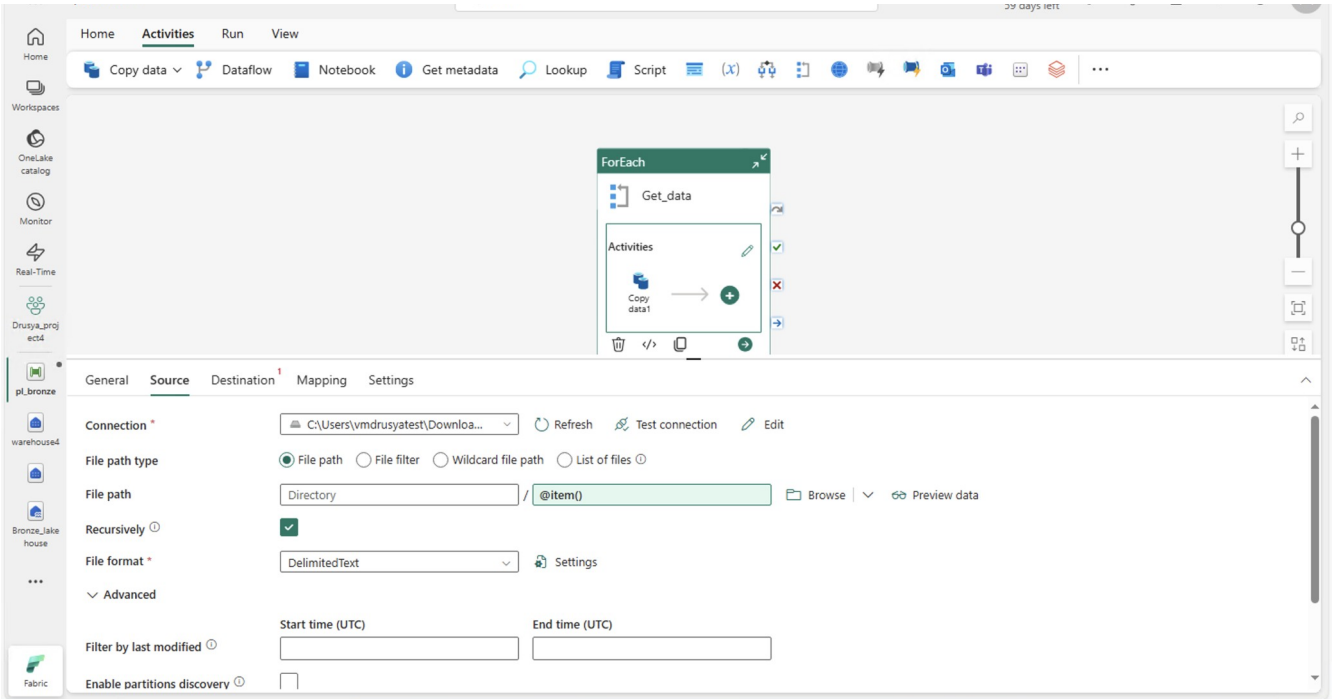


Now add a copy data activity inside the for each and add the source as on-prem data gateway and destination as lakehouse tables.

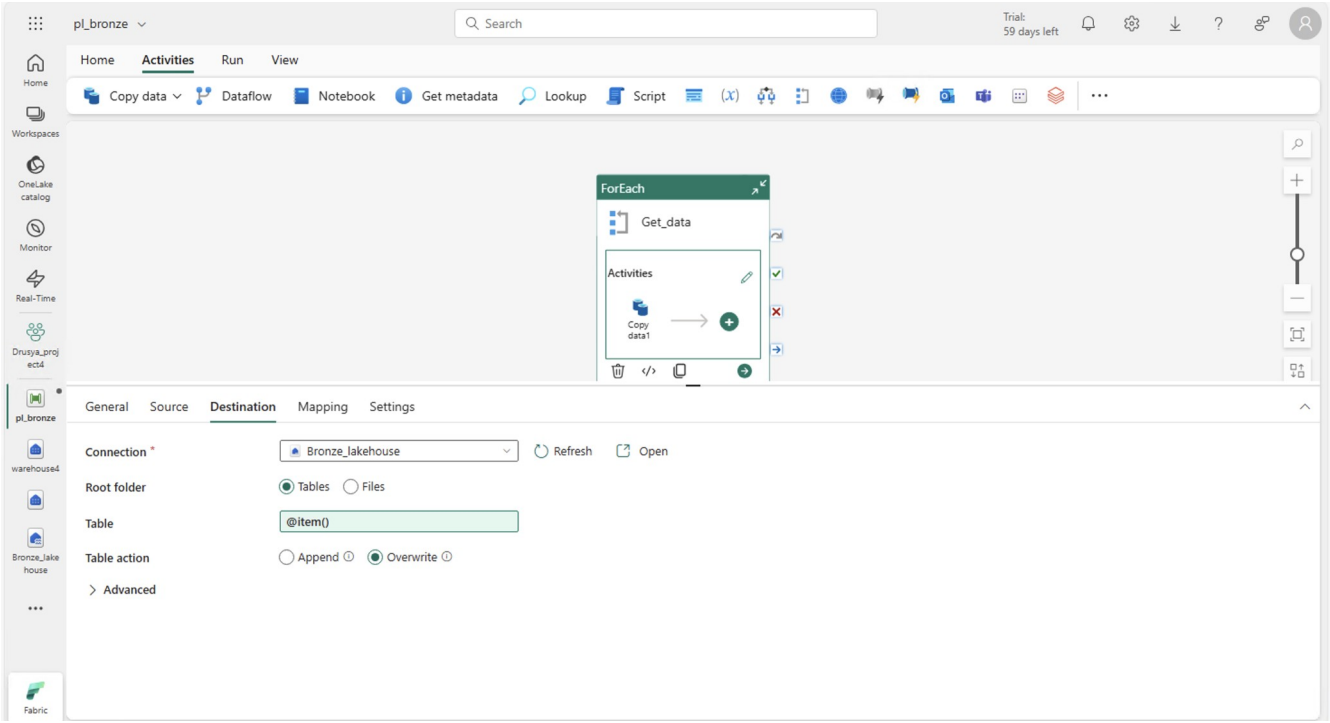
Created a new connection as data gateway in the source side.



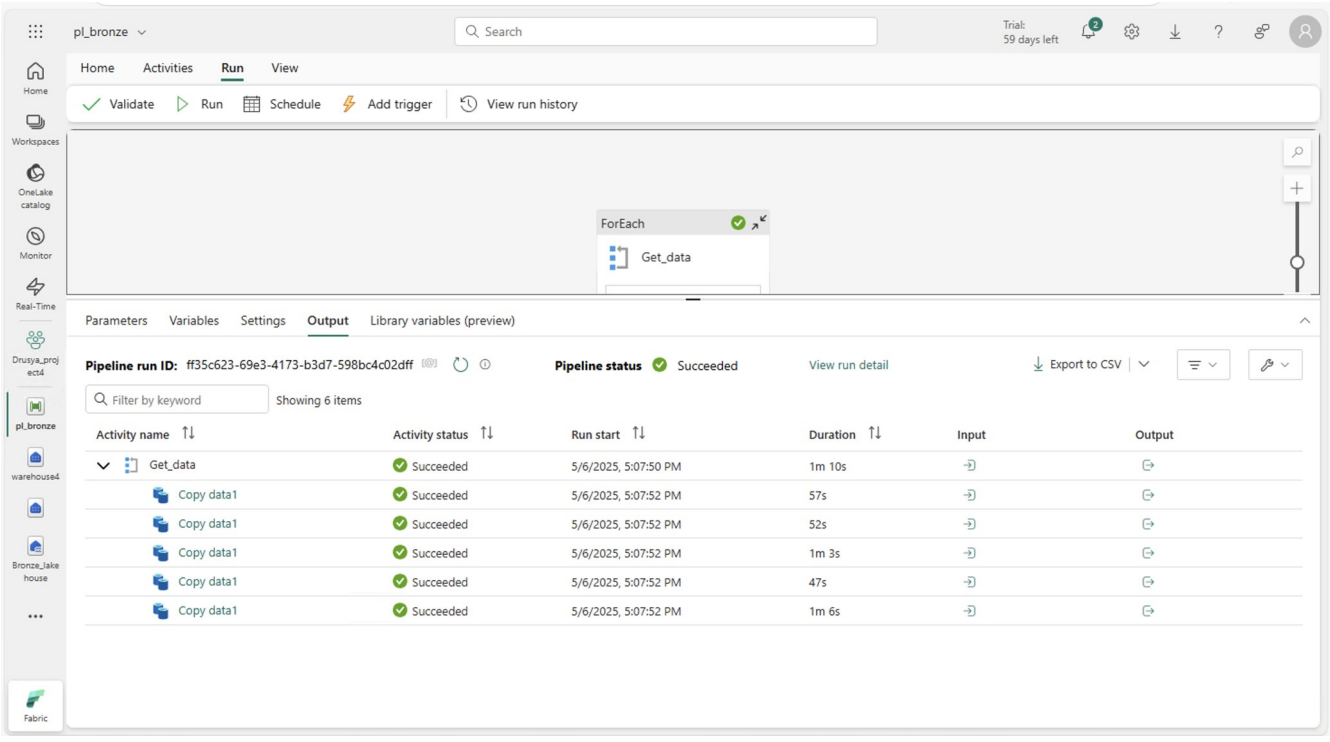
Added the source details.

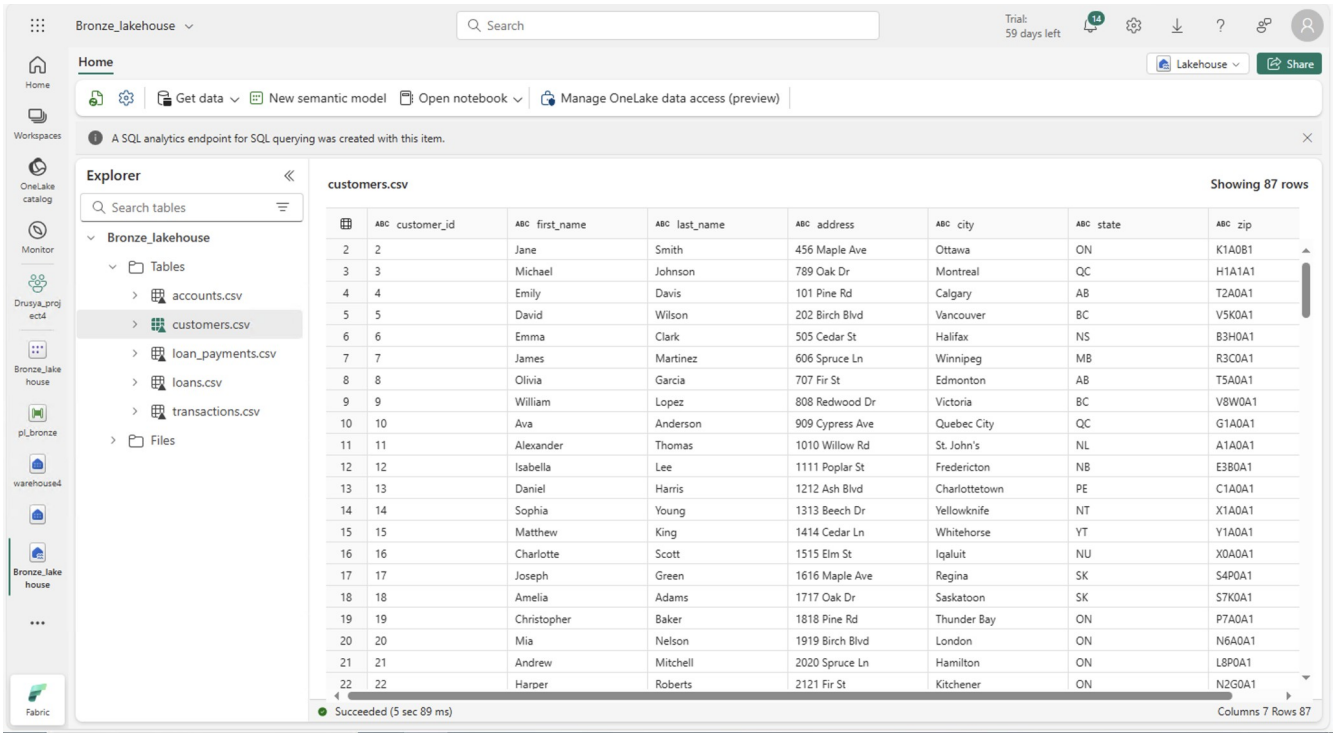


Added the destination details.



Run the pipeline and confirmed the data is loaded into lakehouse tables.



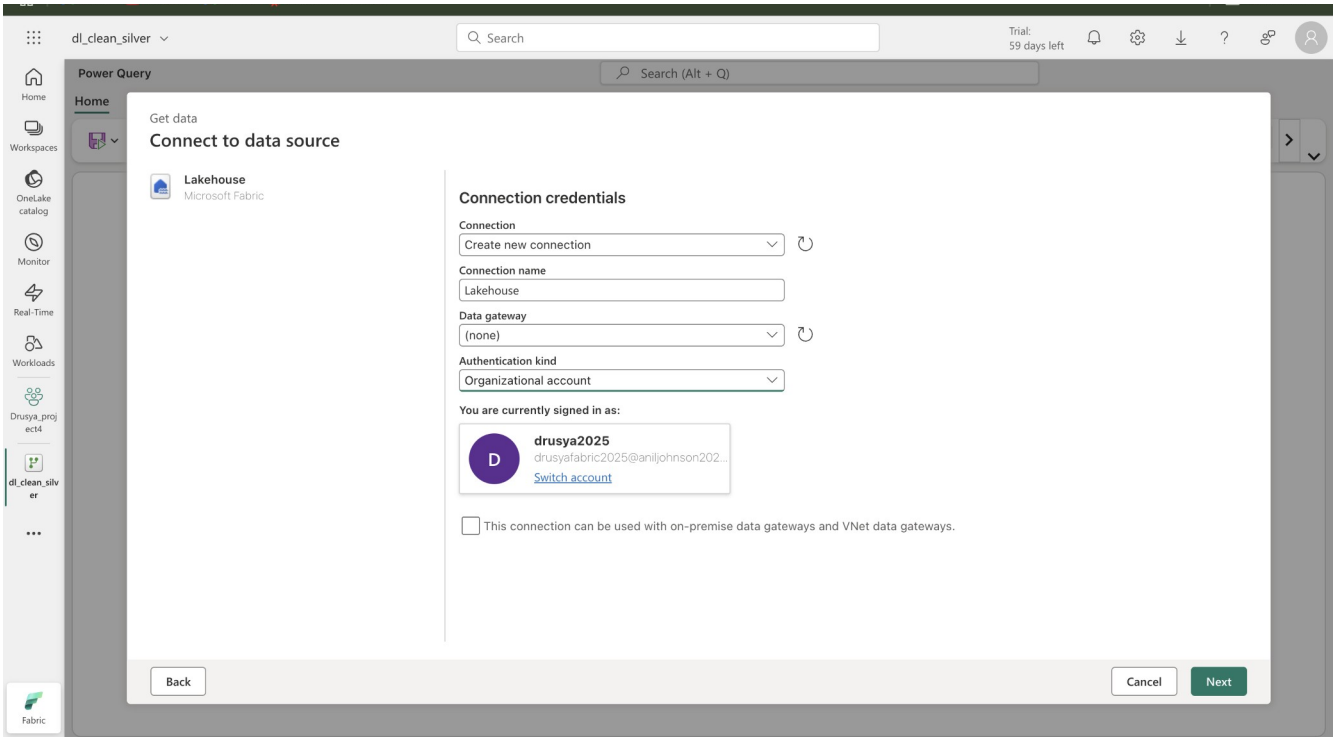


	customer_id	first_name	last_name	address	city	state	zip
2	2	Jane	Smith	456 Maple Ave	Ottawa	ON	K1A0B1
3	3	Michael	Johnson	789 Oak Dr	Montreal	QC	H1A1A1
4	4	Emily	Davis	101 Pine Rd	Calgary	AB	T2A0A1
5	5	David	Wilson	202 Birch Blvd	Vancouver	BC	V5K0A1
6	6	Emma	Clark	505 Cedar St	Halifax	NS	B3H0A1
7	7	James	Martinez	606 Spruce Ln	Winnipeg	MB	R3C0A1
8	8	Olivia	Garcia	707 Fir St	Edmonton	AB	T5A0A1
9	9	William	Lopez	808 Redwood Dr	Victoria	BC	V8W0A1
10	10	Ava	Anderson	909 Cypress Ave	Quebec City	QC	G1A0A1
11	11	Alexander	Thomas	1010 Willow Rd	St. John's	NL	A1A0A1
12	12	Isabella	Lee	1111 Poplar St	Fredericton	NB	E3B0A1
13	13	Daniel	Harris	1212 Ash Blvd	Charlottetown	PE	C1A0A1
14	14	Sophia	Young	1313 Beech Dr	Yellowknife	NT	X1A0A1
15	15	Matthew	King	1414 Cedar Ln	Whitehorse	YT	Y1A0A1
16	16	Charlotte	Scott	1515 Elm St	Iqaluit	NU	X0A0A1
17	17	Joseph	Green	1616 Maple Ave	Regina	SK	S4P0A1
18	18	Amelia	Adams	1717 Oak Dr	Saskatoon	SK	S7K0A1
19	19	Christopher	Baker	1818 Pine Rd	Thunder Bay	ON	P7A0A1
20	20	Mia	Nelson	1919 Birch Blvd	London	ON	N6A0A1
21	21	Andrew	Mitchell	2020 Spruce Ln	Hamilton	ON	L8P0A1
22	22	Harper	Roberts	2121 Fir St	Kitchener	ON	N2G0A1

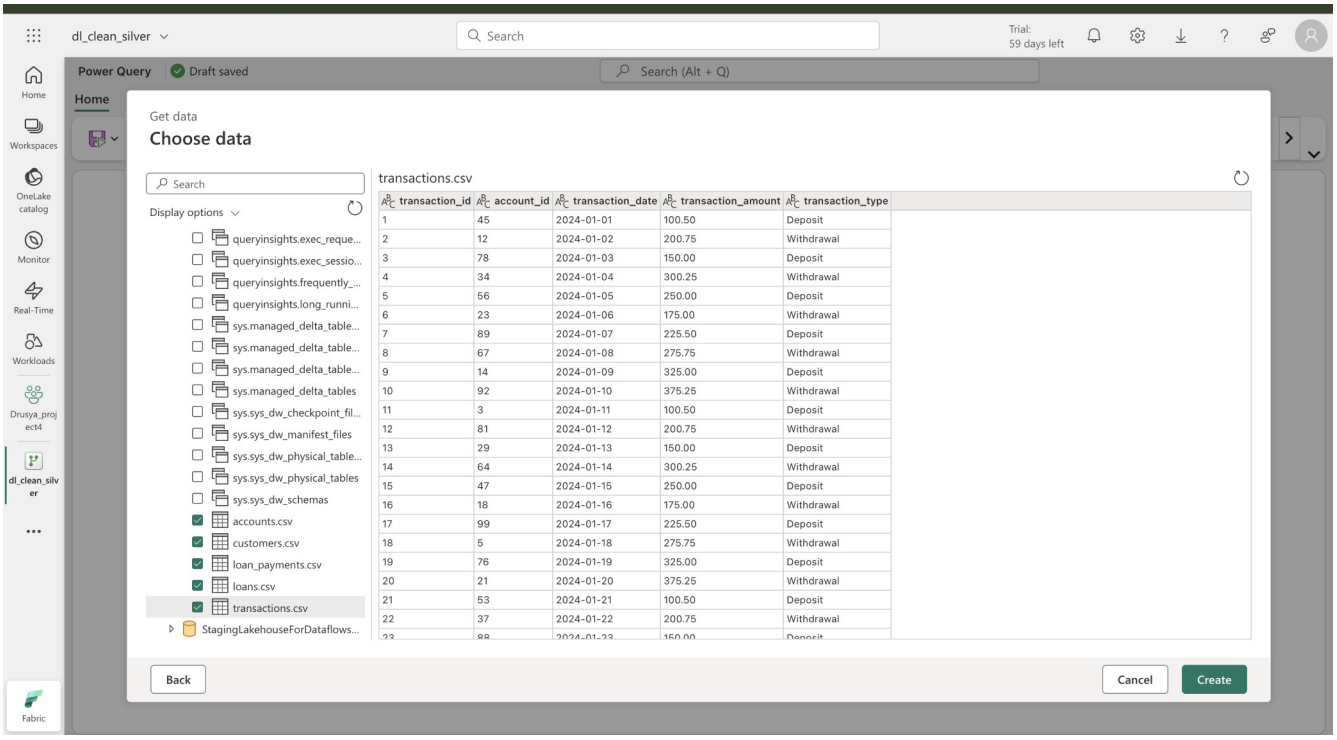
CLEANING THE DATA AND LOADING INTO WAREHOUSE – SILVER LAYER

Now we have our data in the lakehouse we have to clean the data before any transformation for that we are using the data flow activity.

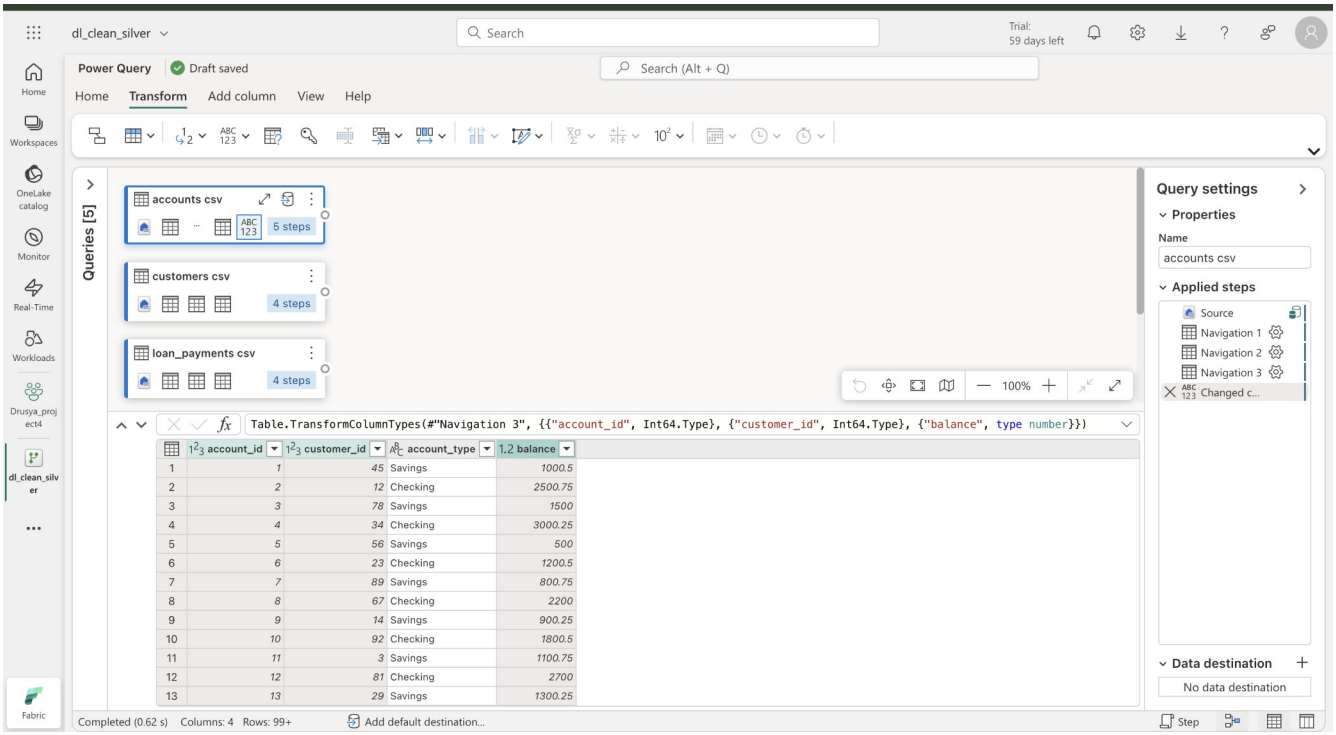
Added the source as lakehouse.



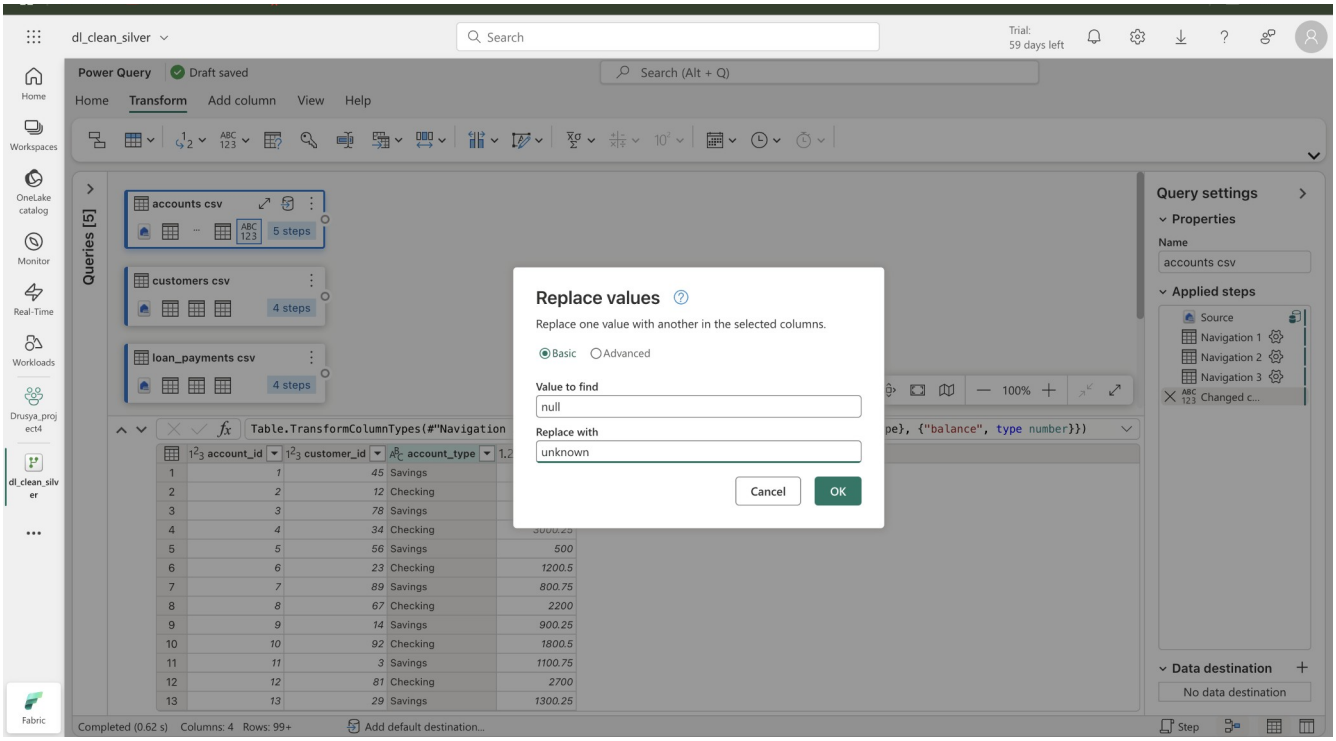
Selected the tables I want to clean.



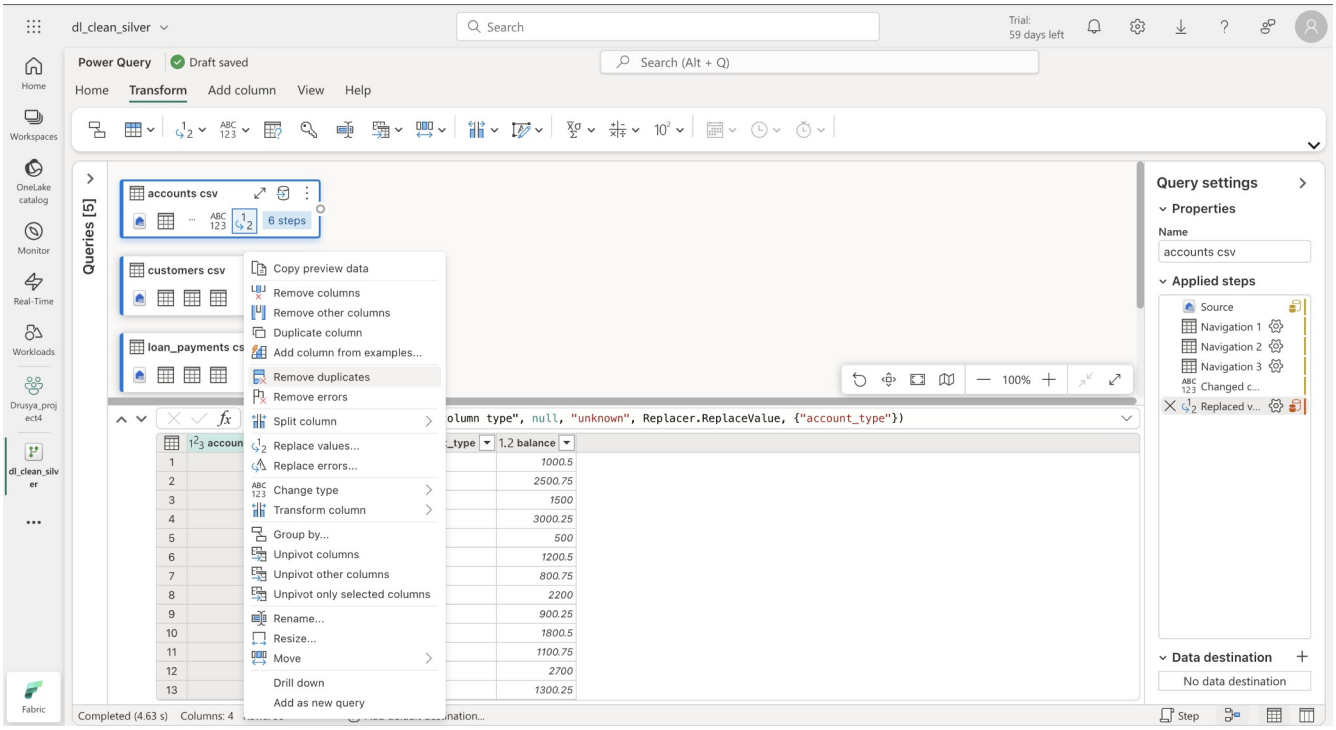
The transformations I used are data type changed.



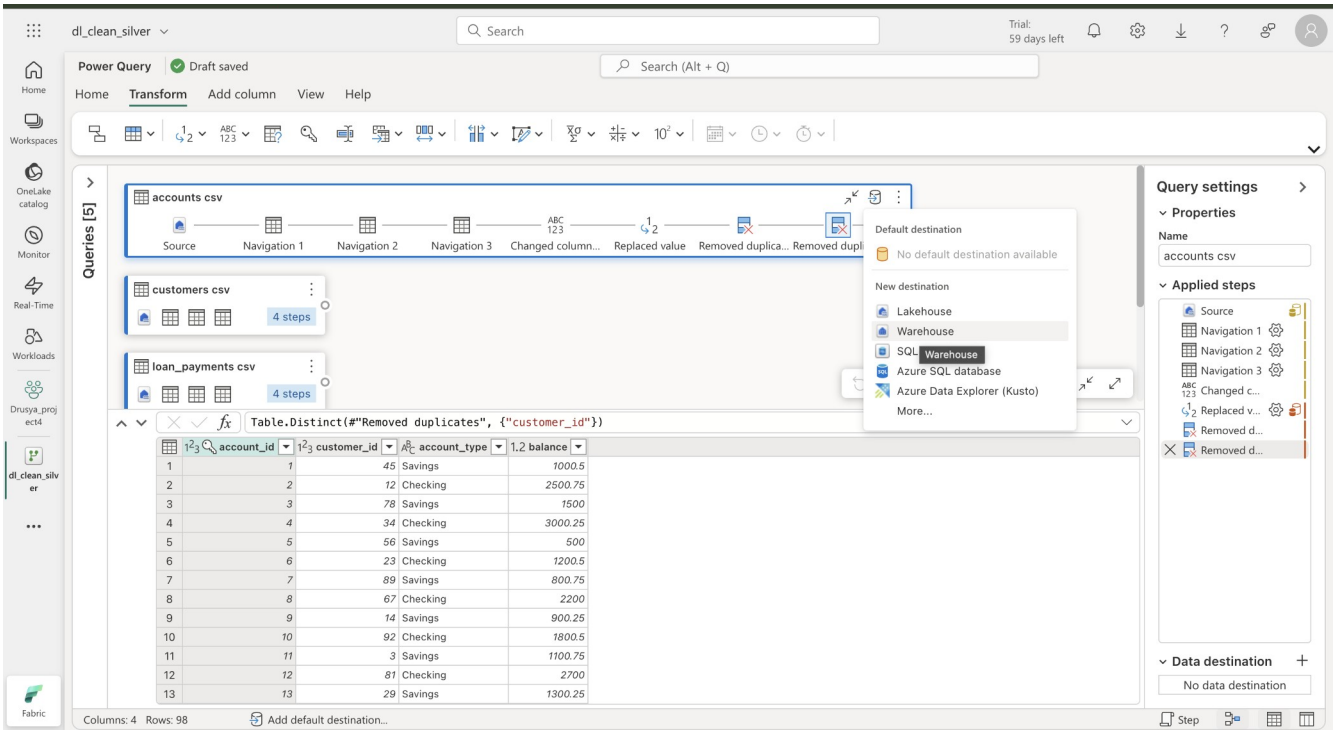
Replaced null with unknown values.

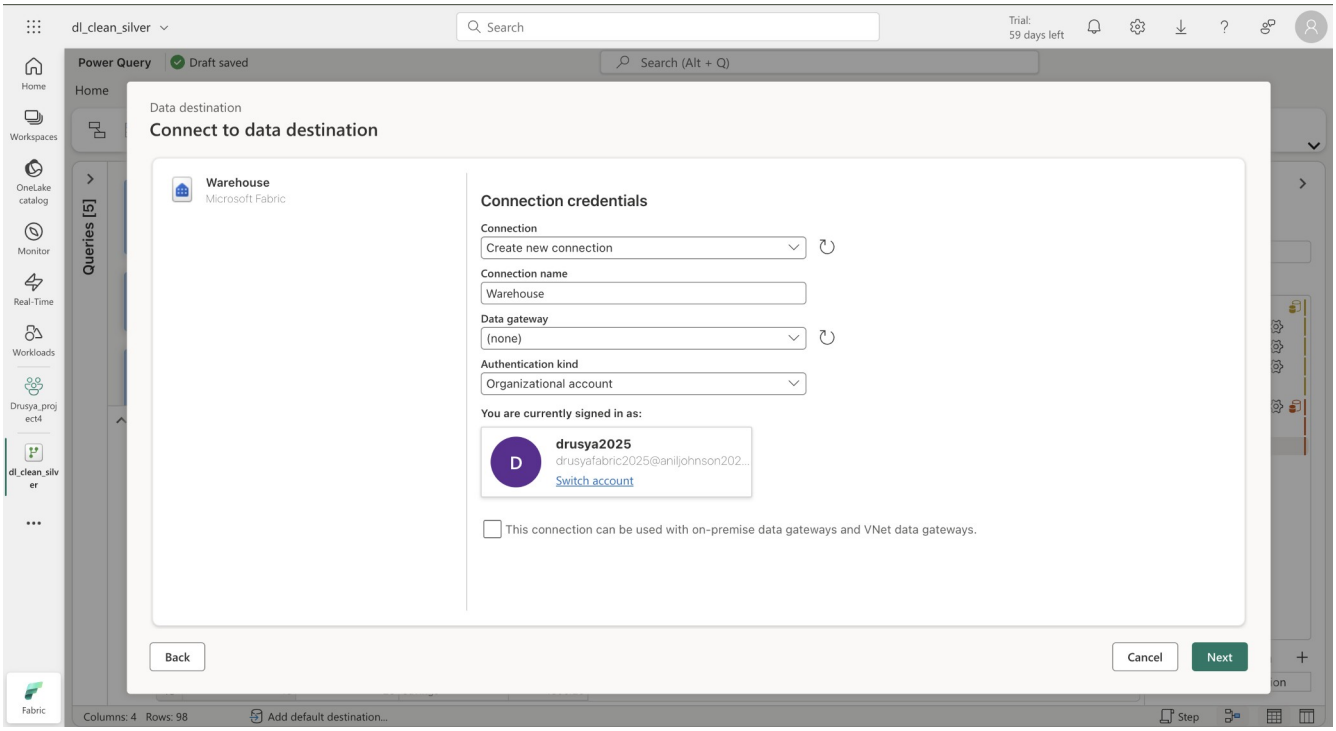


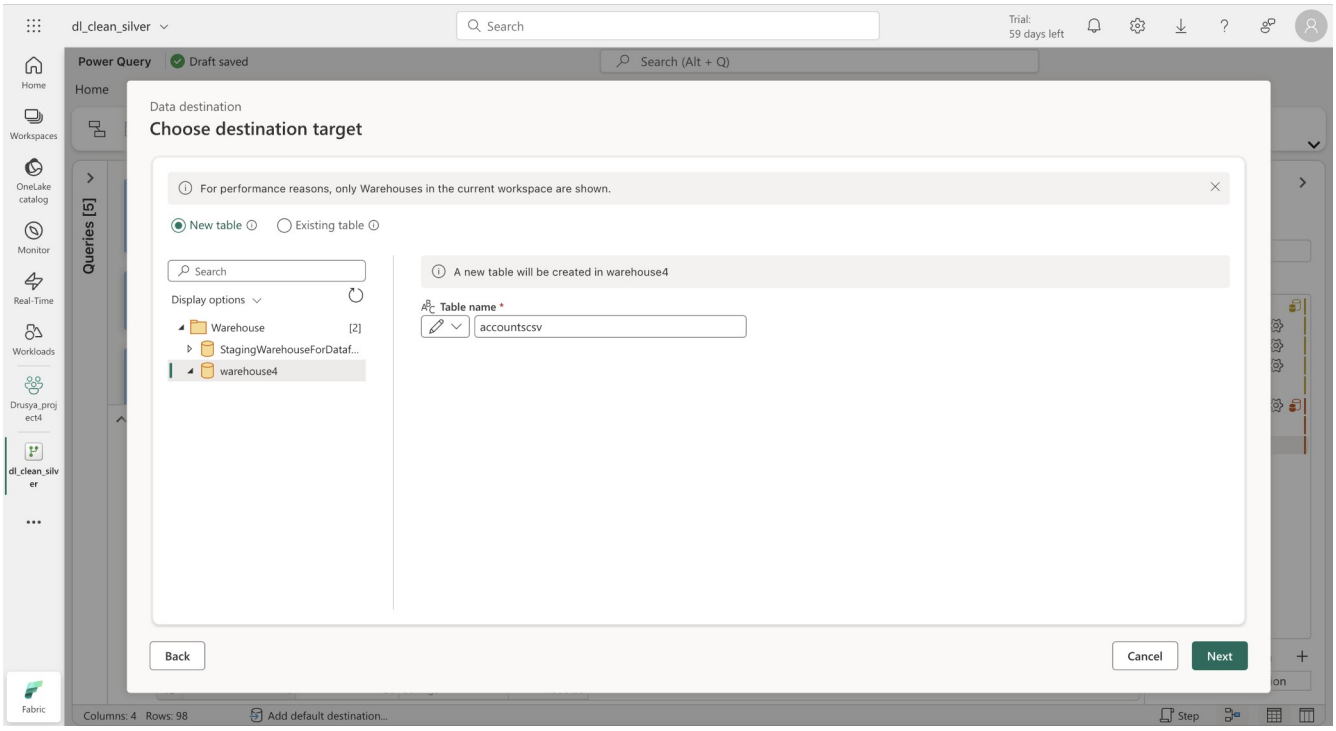
Removed duplicates.

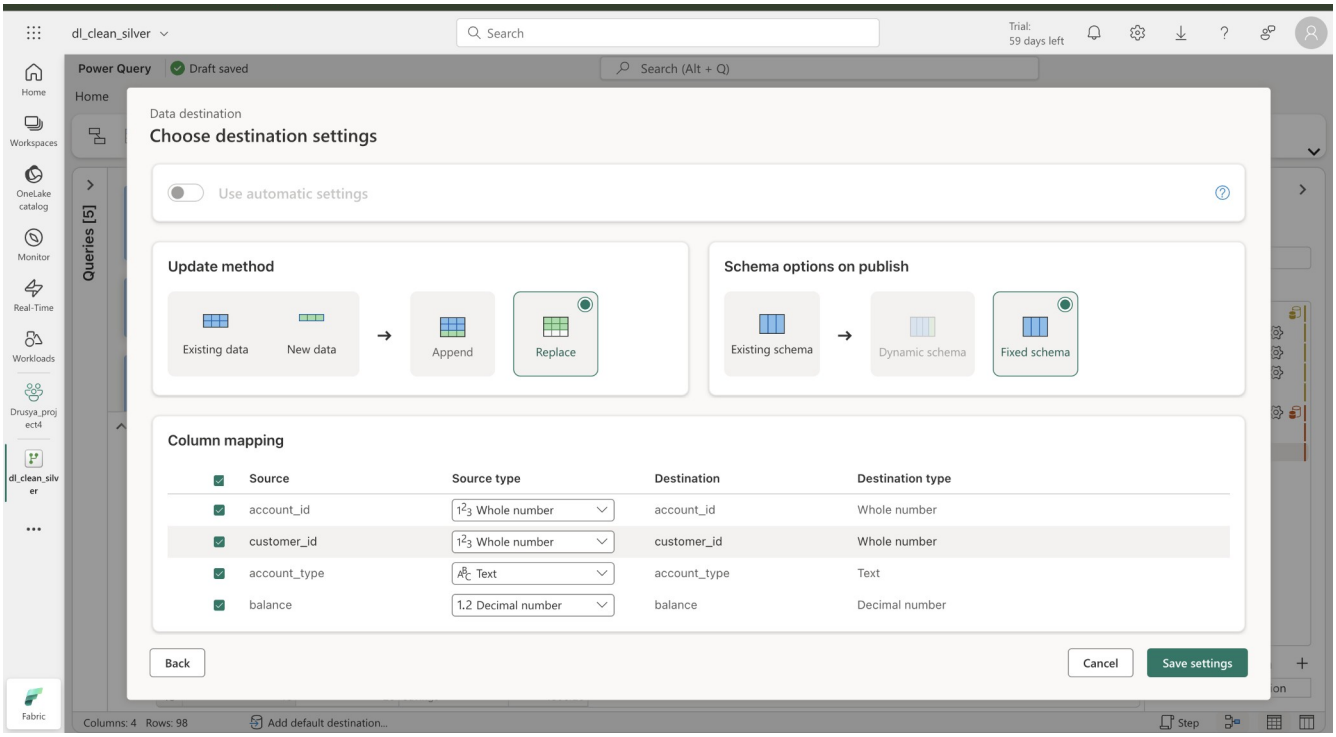


Add destination as warehouse.

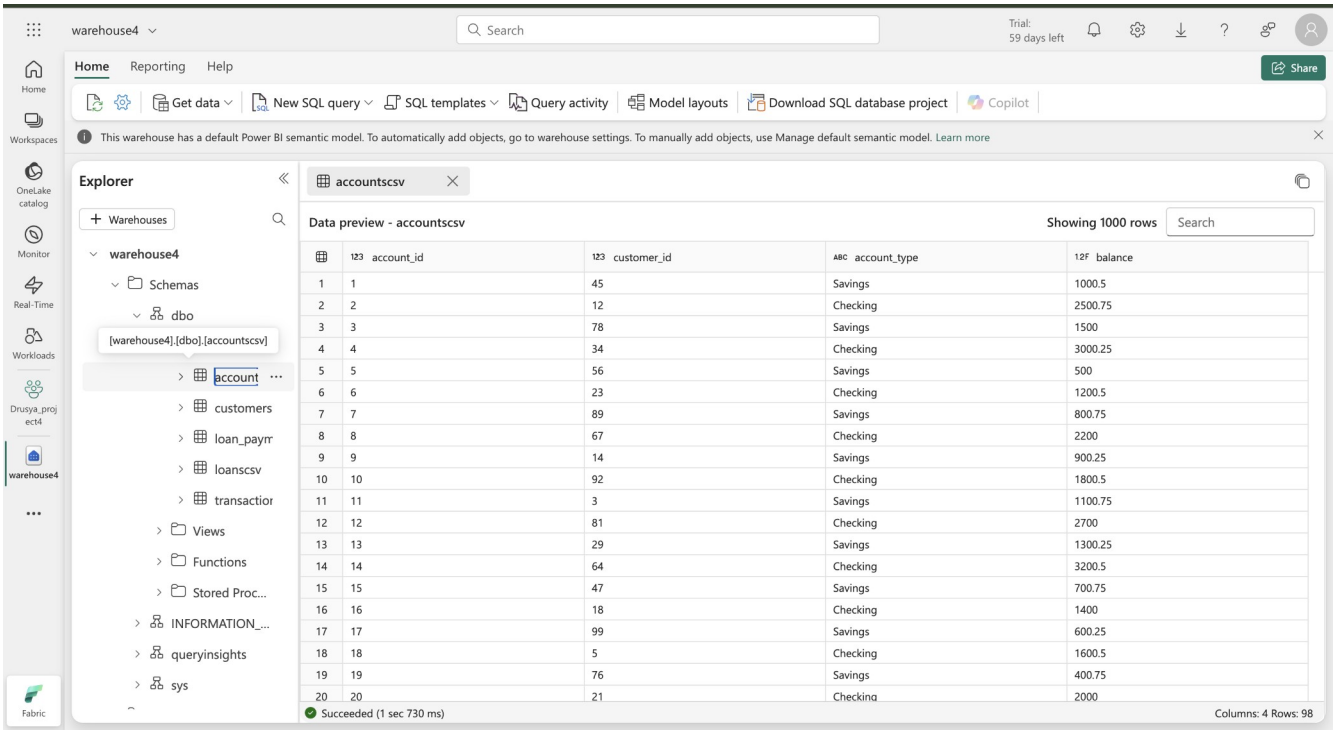


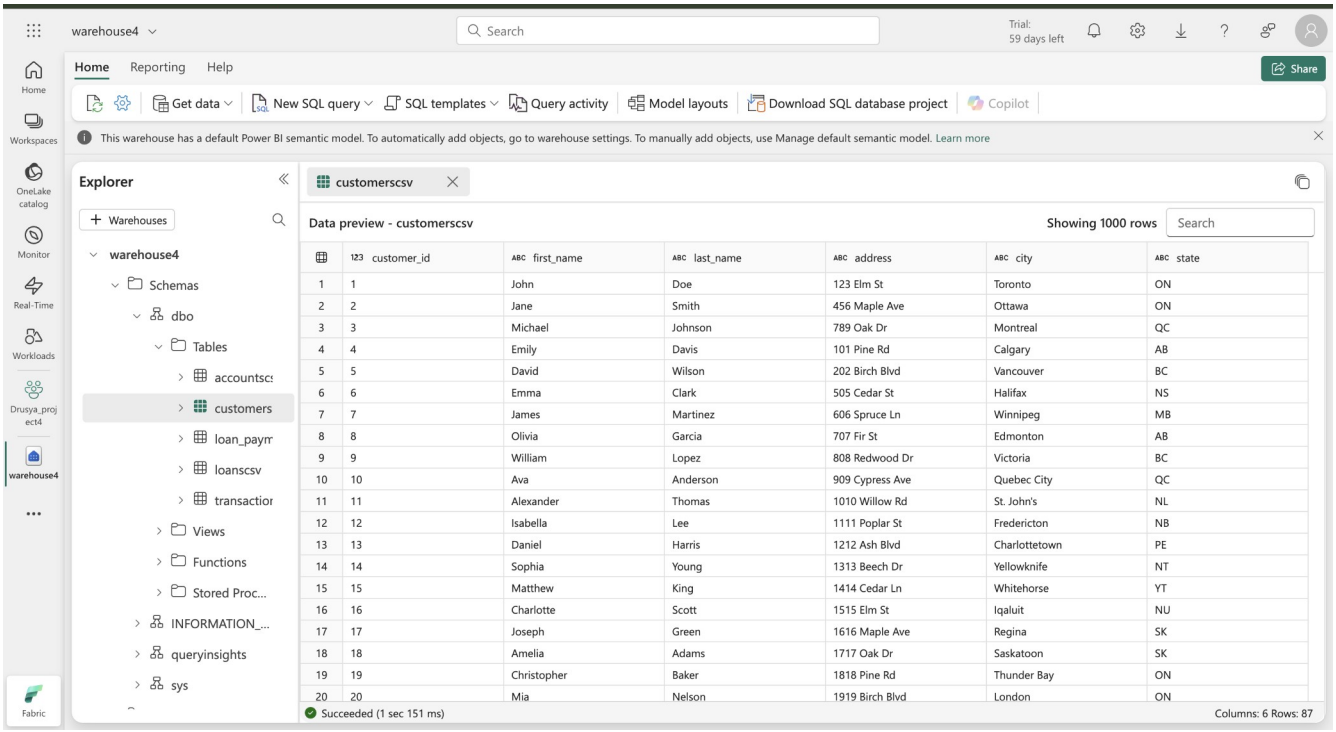

dl_clean_silver
Search
Trial: 59 days left
Power Query
Draft saved
Search (Alt + Q)
Home
Data destination
Connect to data destination
Warehouse
Microsoft Fabric
Connection credentials
Connection
Create new connection
Connection name
Warehouse
Data gateway
(none)
Authentication kind
Organizational account
You are currently signed in as:
D
drusya2025
drusyafabric2025@aniljohnson202...
Switch account
This connection can be used with on-premise data gateways and VNet data gateways.
Back
Cancel
Next
Queries [5]
Columns: 4 Rows: 98
Add default destination...
Step

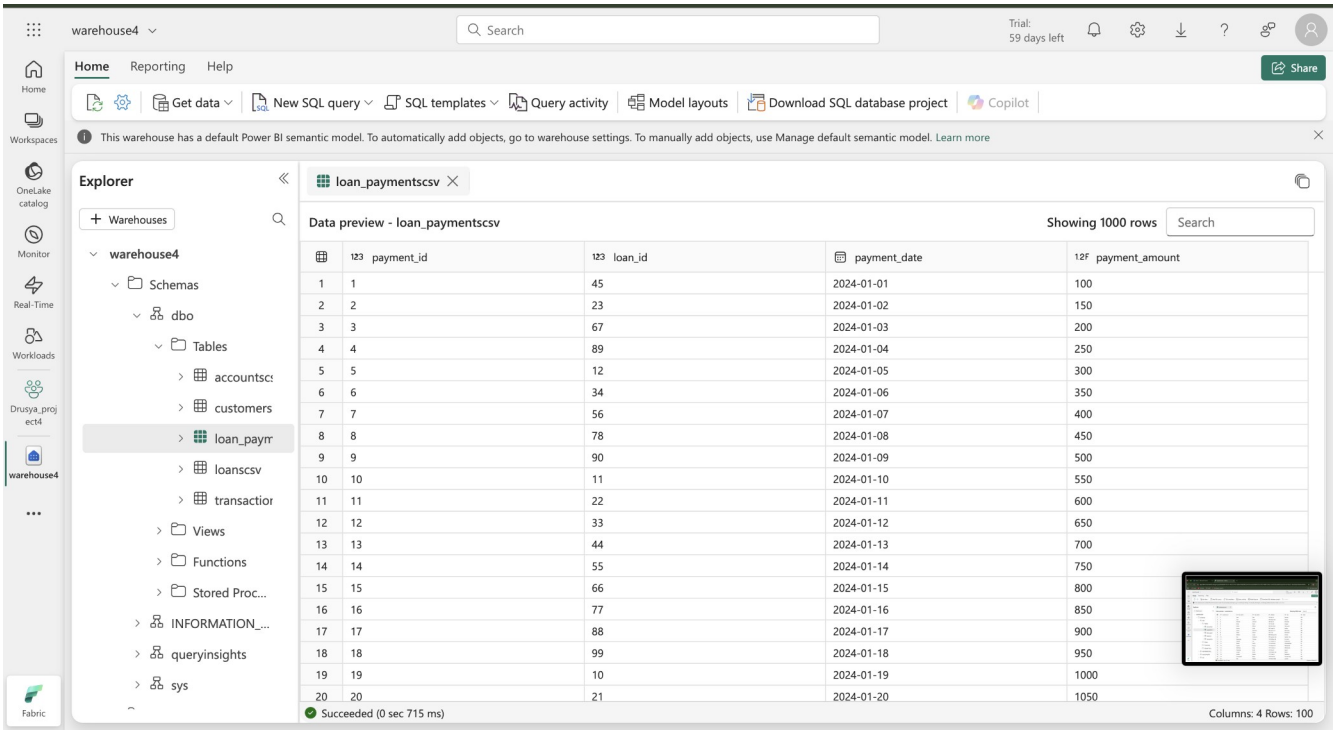

dl_clean_silver
Search
Trial: 59 days left
Power Query
Draft saved
Search (Alt + Q)
Home
Data destination
Choose destination target
For performance reasons, only Warehouses in the current workspace are shown.
New table
Existing table
Search
Display options
Warehouse [2]
StagingWarehouseForDataf...
warehouse4
A new table will be created in warehouse4
Table name *
accountscsv
Back
Cancel
Next
Queries [5]
Columns: 4 Rows: 98
Add default destination...
Step

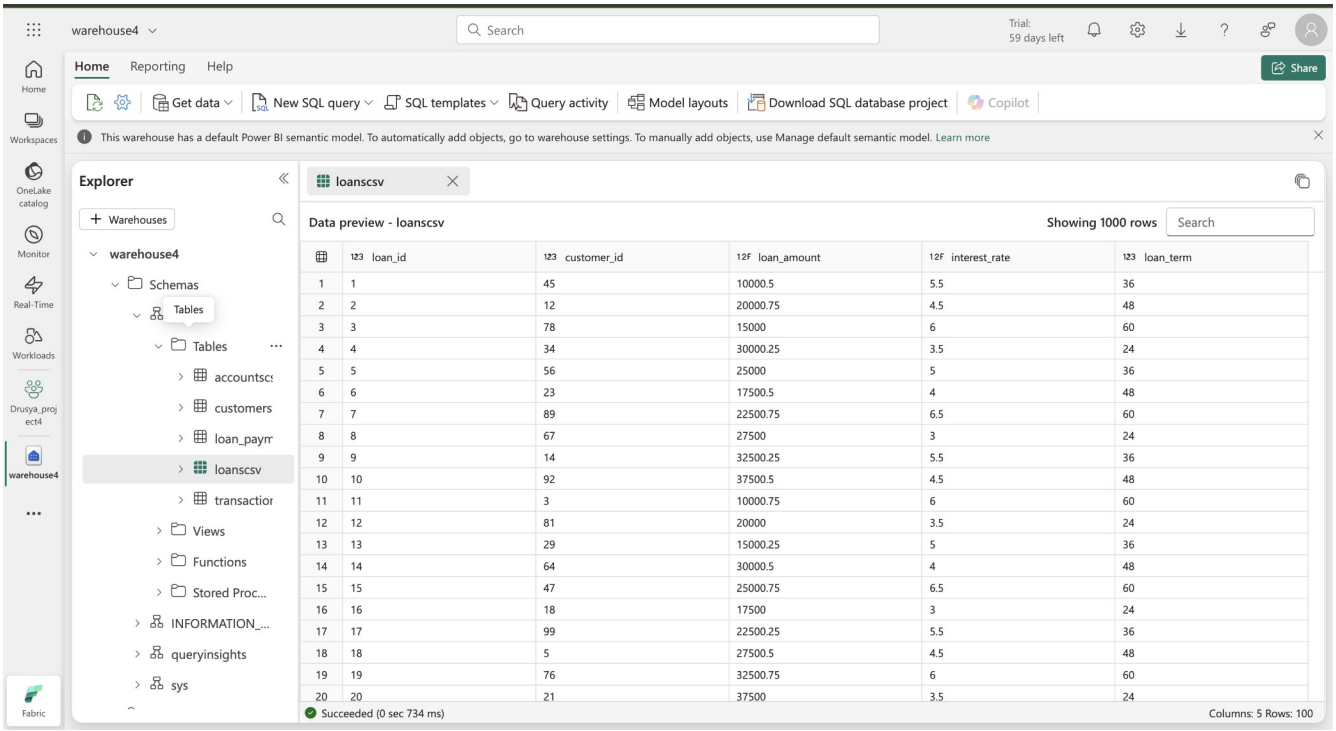


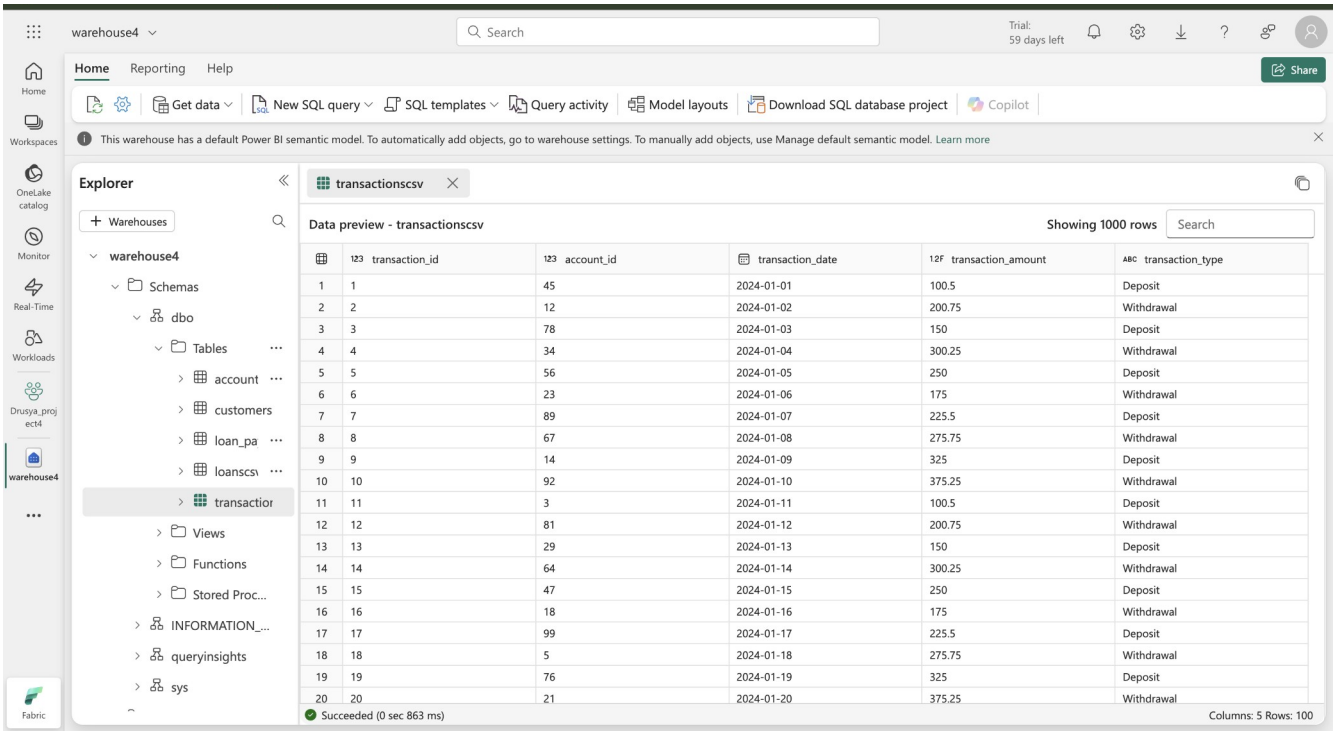
The same process repeated for all tables and confirmed the cleaned data is loaded in the warehouse as tables.




warehouse4
Search
Trial: 59 days left
Home
Reporting
Help
Share
Get data
New SQL query
SQL templates
Query activity
Model layouts
Download SQL database project
Copilot
This warehouse has a default Power BI semantic model. To automatically add objects, go to warehouse settings. To manually add objects, use Manage default semantic model. Learn more
Explorer
customerscsv
Warehouses
warehouse4
Schemas
dbo
Tables
accountsc
customers
loan_paym
loanscsv
transactior
Views
Functions
Stored Proc...
INFORMATION_...
queryinsights
sys
Data preview - customerscsv
Showing 1000 rows
Search
customer_id
first_name
last_name
address
city
state
1 1 John Doe 123 Elm St Toronto ON
2 2 Jane Smith 456 Maple Ave Ottawa ON
3 3 Michael Johnson 789 Oak Dr Montreal QC
4 4 Emily Davis 101 Pine Rd Calgary AB
5 5 David Wilson 202 Birch Blvd Vancouver BC
6 6 Emma Clark 505 Cedar St Halifax NS
7 7 James Martinez 606 Spruce Ln Winnipeg MB
8 8 Olivia Garcia 707 Fir St Edmonton AB
9 9 William Lopez 808 Redwood Dr Victoria BC
10 10 Ava Anderson 909 Cypress Ave Quebec City QC
11 11 Alexander Thomas 1010 Willow Rd St. John's NL
12 12 Isabella Lee 1111 Poplar St Fredericton NB
13 13 Daniel Harris 1212 Ash Blvd Charlottetown PE
14 14 Sophia Young 1313 Beech Dr Yellowknife NT
15 15 Matthew King 1414 Cedar Ln Whitehorse YT
16 16 Charlotte Scott 1515 Elm St Iqaluit NU
17 17 Joseph Green 1616 Maple Ave Regina SK
18 18 Amelia Adams 1717 Oak Dr Saskatoon SK
19 19 Christopher Baker 1818 Pine Rd Thunder Bay ON
20 20 Mia Nelson 1919 Birch Blvd London ON
Succeeded (1 sec 151 ms)
Columns: 6 Rows: 87


warehouse4
Search
Trial: 59 days left
Home
Reporting
Help
Share
Get data
New SQL query
SQL templates
Query activity
Model layouts
Download SQL database project
Copilot
This warehouse has a default Power BI semantic model. To automatically add objects, go to warehouse settings. To manually add objects, use Manage default semantic model. Learn more
Explorer
loan_paymentscsv
Warehouses
warehouse4
Schemas
dbo
Tables
accountsc
customers
loan_paym
loanscsv
transactior
Views
Functions
Stored Proc...
INFORMATION_...
queryinsights
sys
Data preview - loan_paymentscsv
Showing 1000 rows
Search
payment_id
loan_id
payment_date
payment_amount
1 1 45 2024-01-01 100
2 2 23 2024-01-02 150
3 3 67 2024-01-03 200
4 4 89 2024-01-04 250
5 5 12 2024-01-05 300
6 6 34 2024-01-06 350
7 7 56 2024-01-07 400
8 8 78 2024-01-08 450
9 9 90 2024-01-09 500
10 10 11 2024-01-10 550
11 11 22 2024-01-11 600
12 12 33 2024-01-12 650
13 13 44 2024-01-13 700
14 14 55 2024-01-14 750
15 15 66 2024-01-15 800
16 16 77 2024-01-16 850
17 17 88 2024-01-17 900
18 18 99 2024-01-18 950
19 19 10 2024-01-19 1000
20 20 21 2024-01-20 1050
Succeeded (0 sec 715 ms)
Columns: 4 Rows: 100

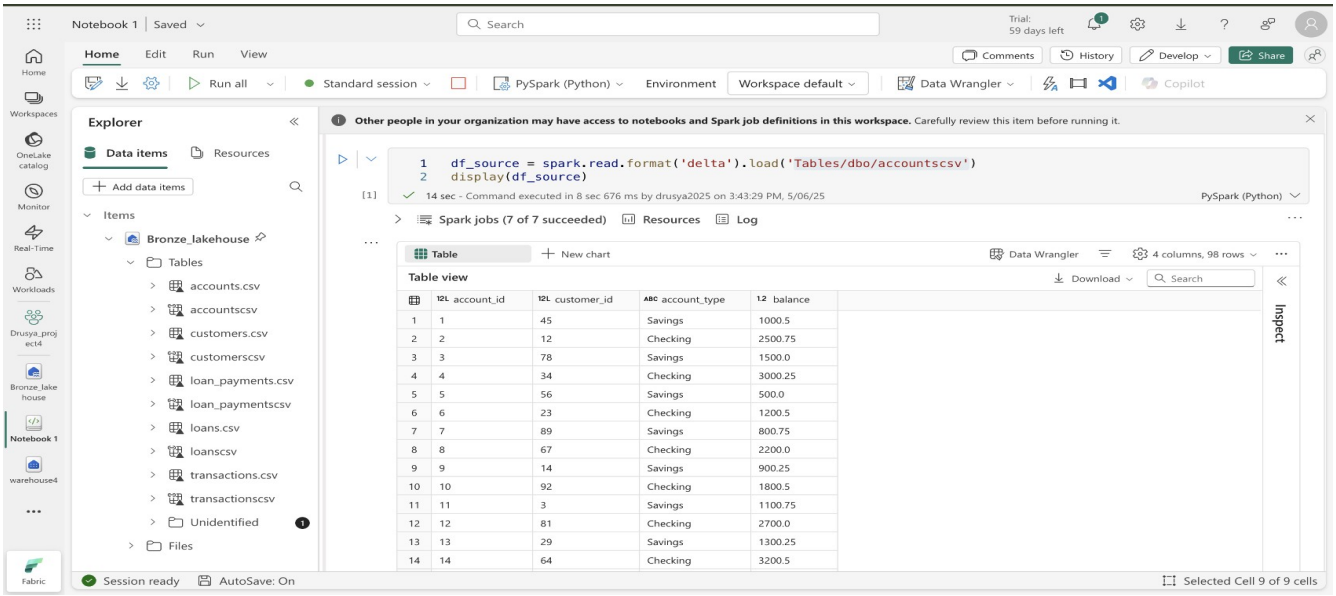

warehouse4
Search
Trial: 59 days left
Home
Reporting
Help
Share
Get data
New SQL query
SQL templates
Query activity
Model layouts
Download SQL database project
Copilot
This warehouse has a default Power BI semantic model. To automatically add objects, go to warehouse settings. To manually add objects, use Manage default semantic model. Learn more
Explorer
loanscsv
Warehouses
warehouse4
Schemas
Tables
accountsc
customers
loan_paym
loanscsv
transactior
Views
Functions
Stored Proc...
INFORMATION_...
queryinsights
sys
Data preview - loanscsv
Showing 1000 rows
Search
loan_id
customer_id
loan_amount
interest_rate
loan_term
1 1 45 10000.5 5.5 36
2 2 12 20000.75 4.5 48
3 3 78 15000 6 60
4 4 34 30000.25 3.5 24
5 5 56 25000 5 36
6 6 23 17500.5 4 48
7 7 89 22500.75 6.5 60
8 8 67 27500 3 24
9 9 14 32500.25 5.5 36
10 10 92 37500.5 4.5 48
11 11 3 10000.75 6 60
12 12 81 20000 3.5 24
13 13 29 15000.25 5 36
14 14 64 30000.5 4 48
15 15 47 25000.75 6.5 60
16 16 18 17500 3 24
17 17 99 22500.25 5.5 36
18 18 5 27500.5 4.5 48
19 19 76 32500.75 6 60
20 20 21 37500 3.5 24
Succeeded (0 sec 734 ms)
Columns: 5 Rows: 100


warehouse4
Search
Trial: 59 days left
Home
Reporting
Help
Share
Get data
New SQL query
SQL templates
Query activity
Model layouts
Download SQL database project
Copilot
This warehouse has a default Power BI semantic model. To automatically add objects, go to warehouse settings. To manually add objects, use Manage default semantic model. Learn more
Explorer
transactionscsv
Warehouses
warehouse4
Schemas
dbo
Tables
account
customers
loan_pa
loanscsv
transactior
Views
Functions
Stored Proc...
INFORMATION_...
queryinsights
sys
Data preview - transactionscsv
Showing 1000 rows
Search
transaction_id
account_id
transaction_date
transaction_amount
transaction_type
1 1 45 2024-01-01 100.5 Deposit
2 2 12 2024-01-02 200.75 Withdrawal
3 3 78 2024-01-03 150 Deposit
4 4 34 2024-01-04 300.25 Withdrawal
5 5 56 2024-01-05 250 Deposit
6 6 23 2024-01-06 175 Withdrawal
7 7 89 2024-01-07 225.5 Deposit
8 8 67 2024-01-08 275.75 Withdrawal
9 9 14 2024-01-09 325 Deposit
10 10 92 2024-01-10 375.25 Withdrawal
11 11 3 2024-01-11 100.5 Deposit
12 12 81 2024-01-12 200.75 Withdrawal
13 13 29 2024-01-13 150 Deposit
14 14 64 2024-01-14 300.25 Withdrawal
15 15 47 2024-01-15 250 Deposit
16 16 18 2024-01-16 175 Withdrawal
17 17 99 2024-01-17 225.5 Deposit
18 18 5 2024-01-18 275.75 Withdrawal
19 19 76 2024-01-19 325 Deposit
20 20 21 2024-01-20 375.25 Withdrawal
Succeeded (0 sec 863 ms)
Columns: 5 Rows: 100

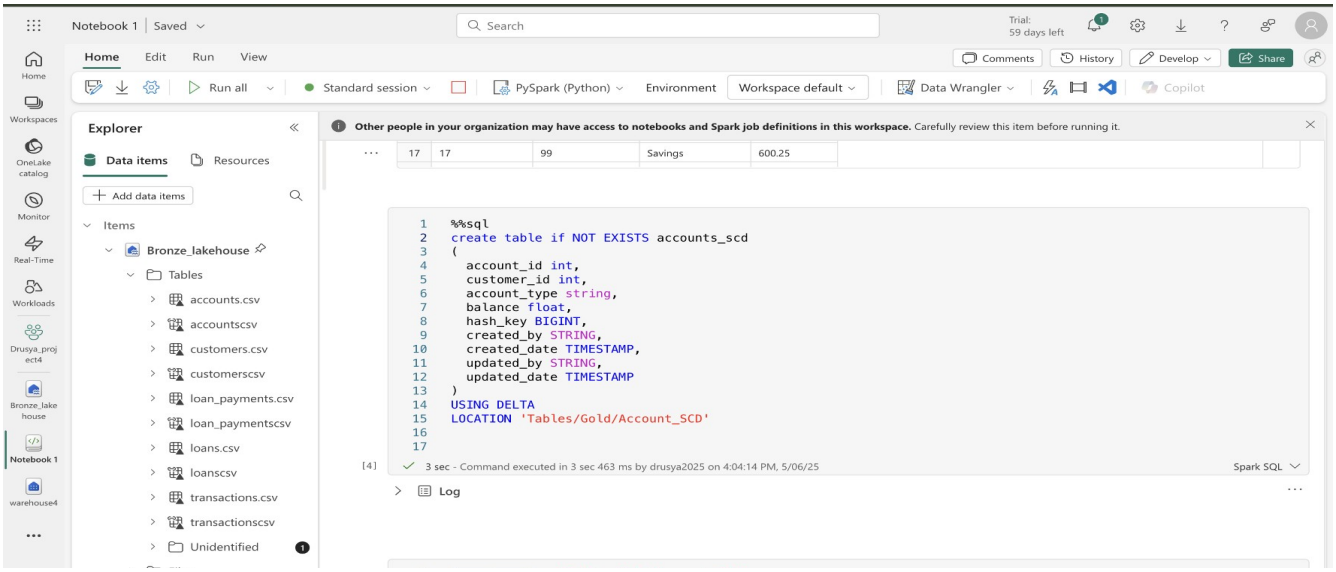
PERFORMING SCD TYPE 1 TRANSFORMATIONS ON THE CLEANED DATA – GOLD LAYER

To perform the SCD Type 1 transformations on the cleaned data we use fabric notebooks here. I used the standard session cluster for running the notebooks and created a short cut to access the data from warehouse.

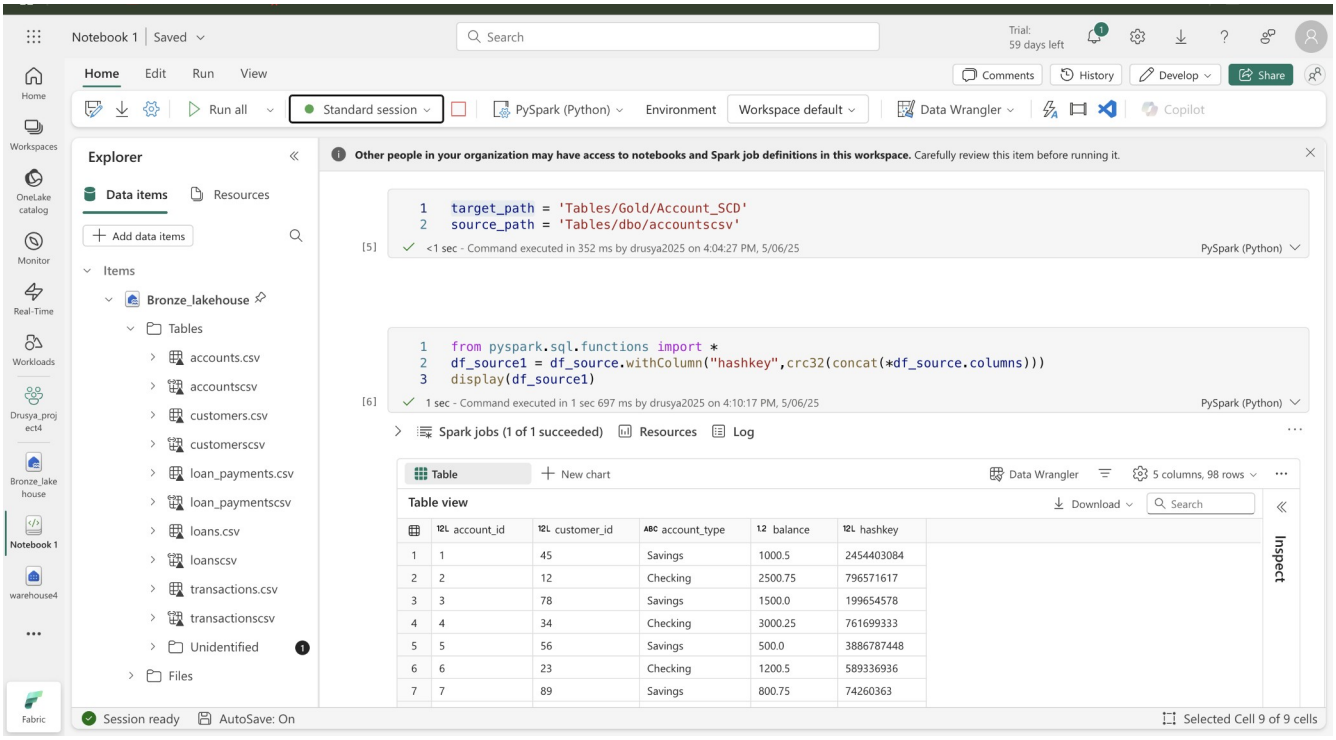
First I read my data from the warehouse table.



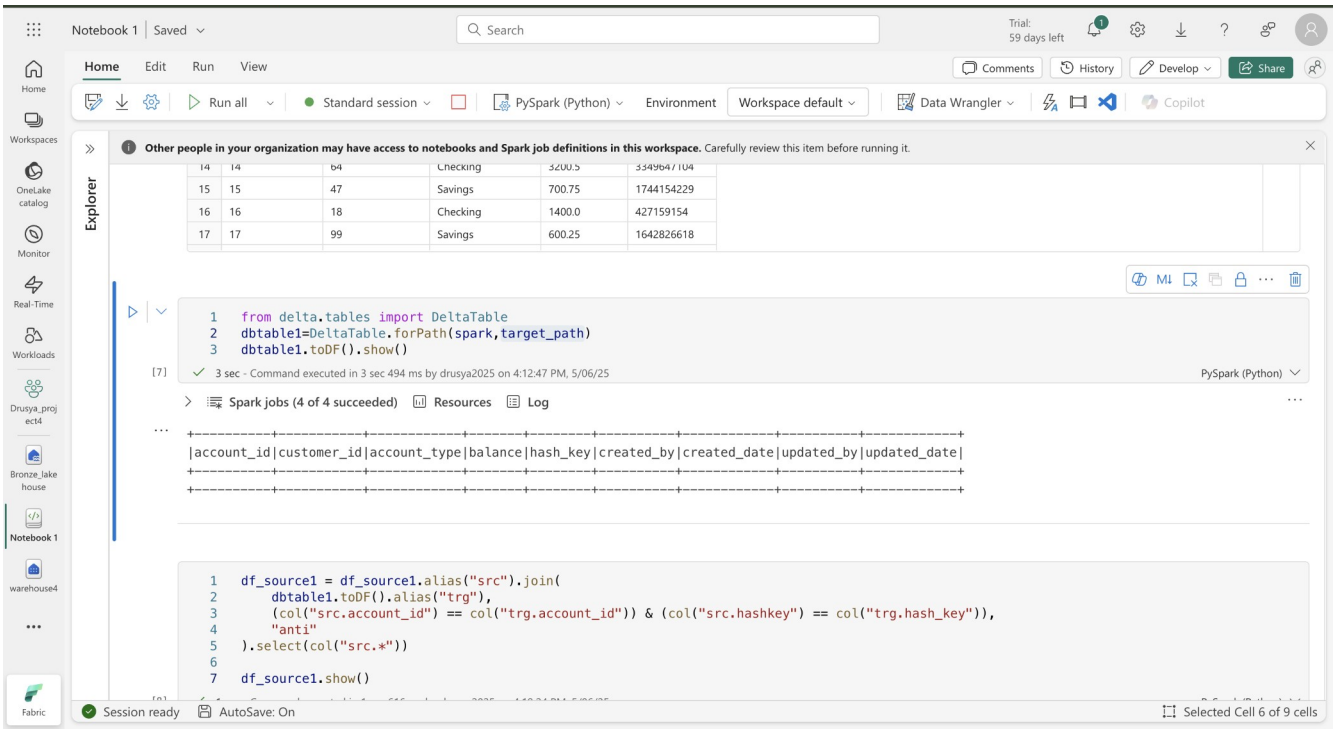
Next I created the delta target table for the file.



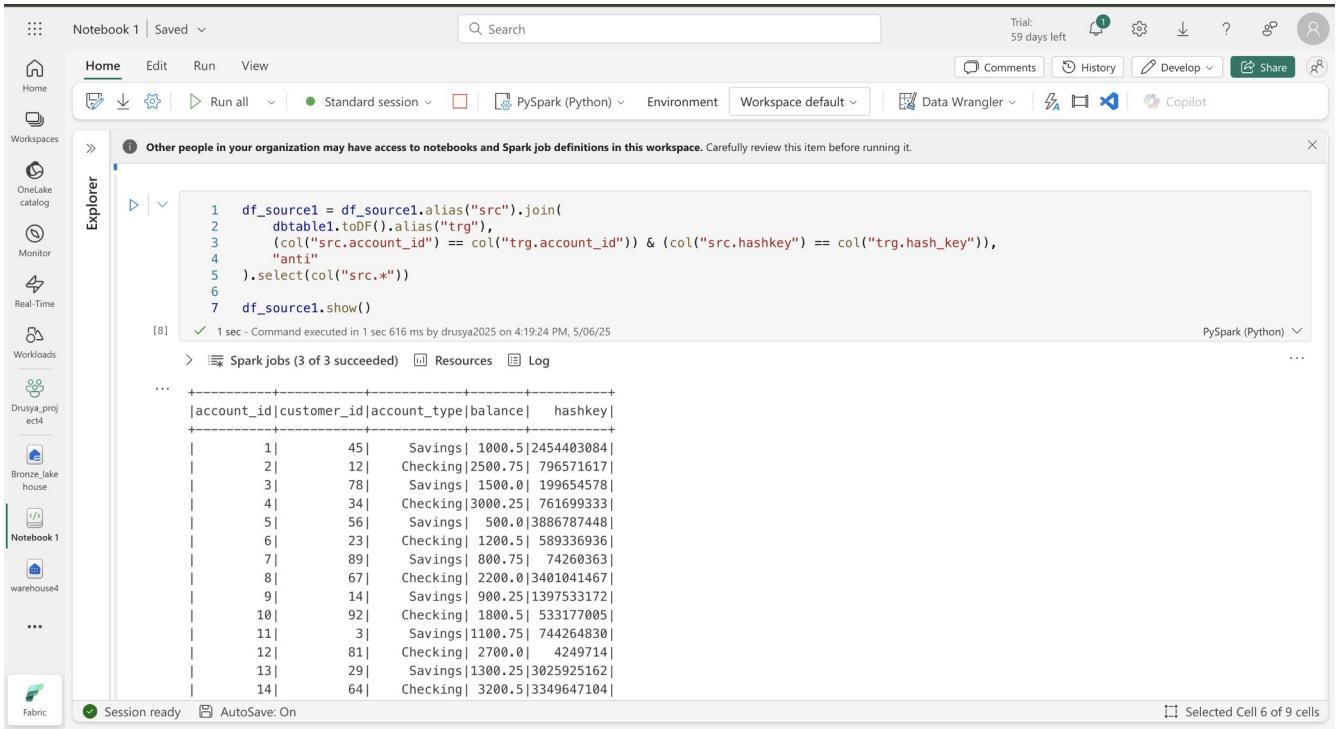
Then I assigned my target file path and source file path, after that I added a new column hashkey to the source file which combines all the column data using the concat and crc32 command and converted it into a data frame.



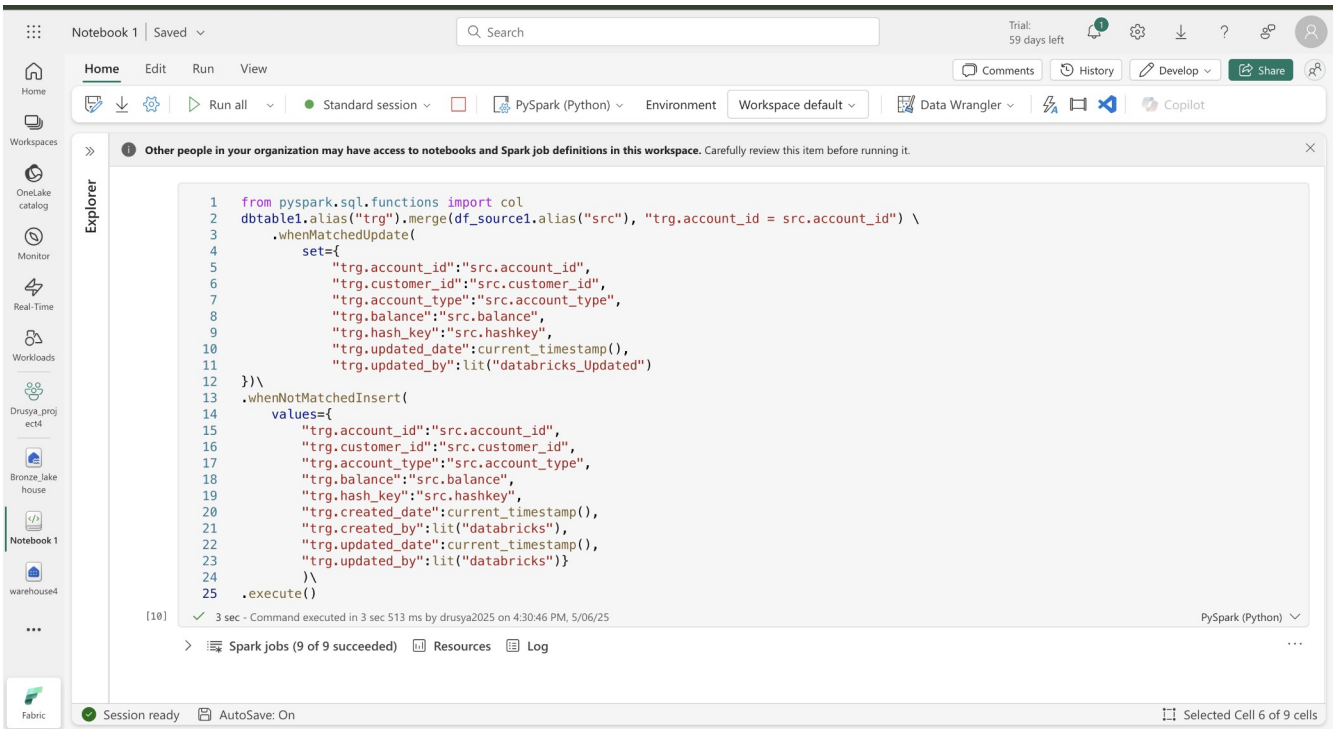
Now I converted my target table into a data frame.



With the help of anti join command I joined my source and target data frame. The purpose of this command is to check the id and hashkey columns so that we can determine the data is an updated one or new.

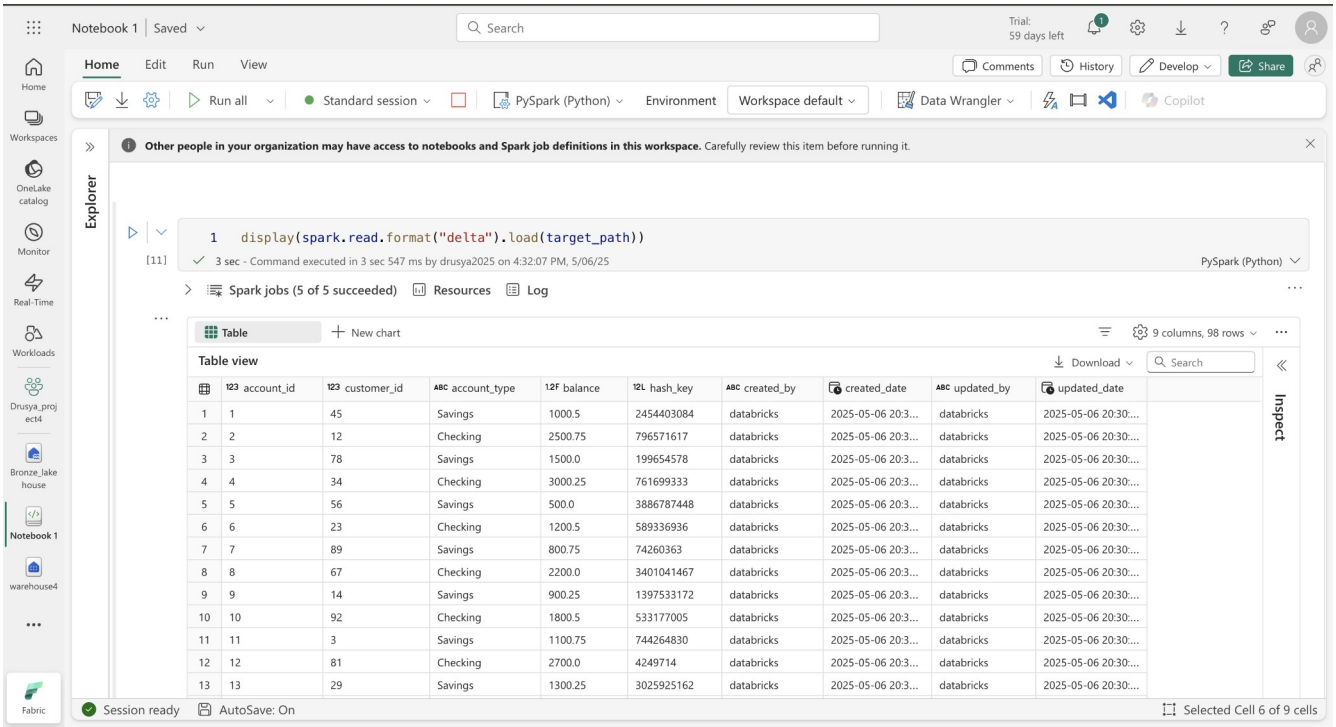


Now with the help of merge command we are updating records based on update or new records into the table.



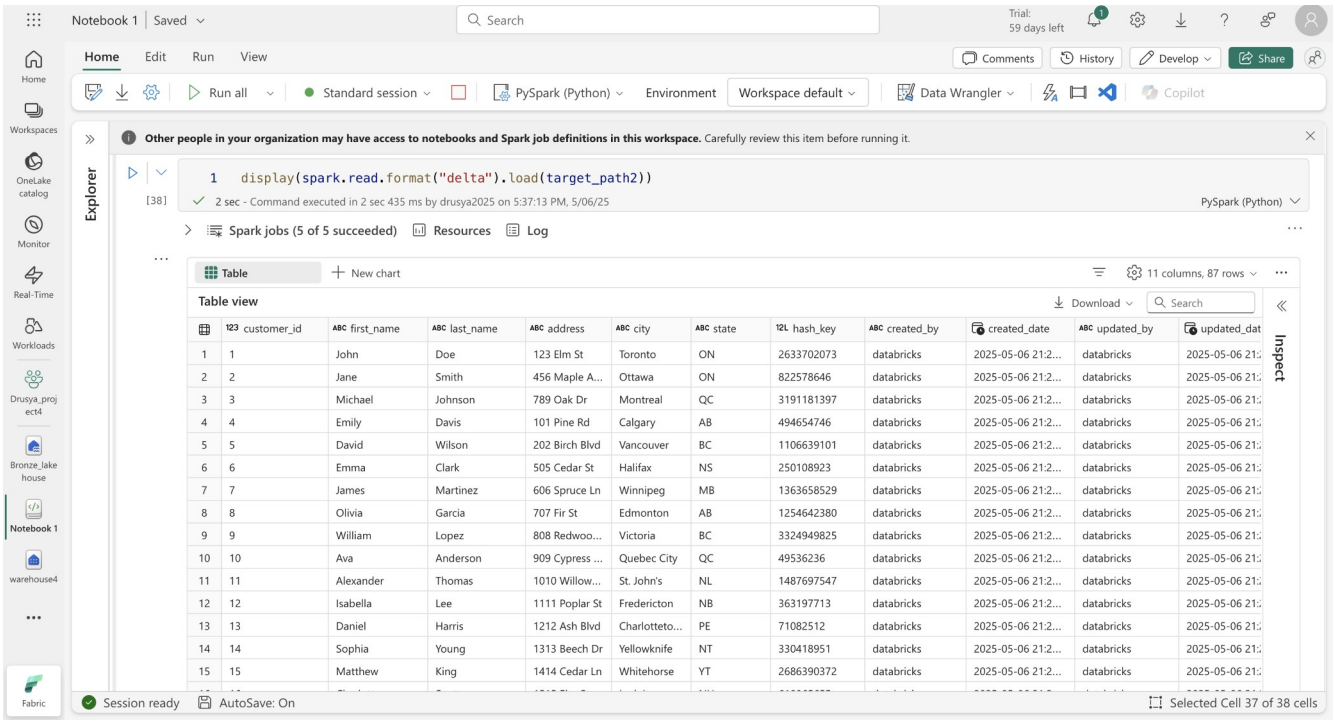
Finally I displayed the output table. Used the same logic on the rest of the files.

Accounts table:



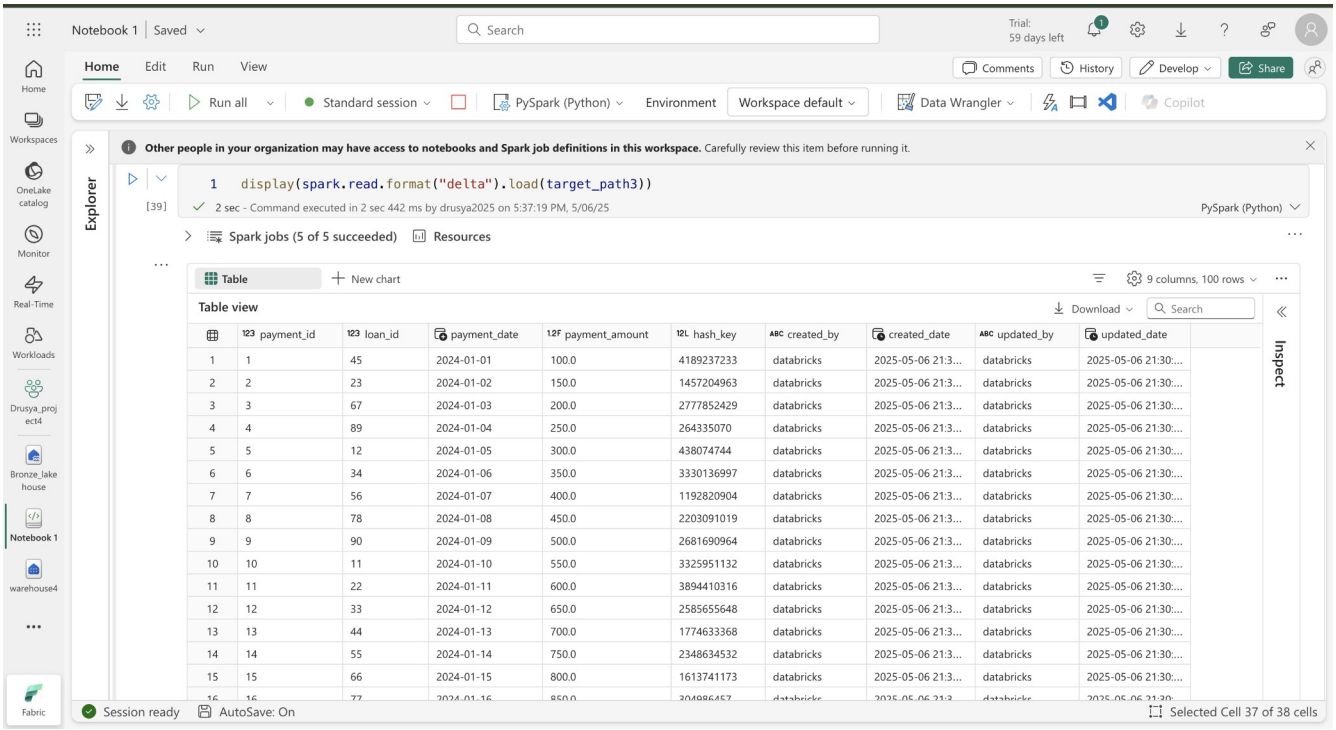
	account_id	customer_id	account_type	balance	hash_key	created_by	created_date	updated_by	updated_date
1	1	45	Savings	1000.5	2454403084	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
2	2	12	Checking	2500.75	796571617	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
3	3	78	Savings	1500.0	199654578	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
4	4	34	Checking	3000.25	761699333	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
5	5	56	Savings	500.0	3886787448	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
6	6	23	Checking	1200.5	589336936	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
7	7	89	Savings	800.75	74260363	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
8	8	67	Checking	2200.0	3401041467	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
9	9	14	Savings	900.25	1397533172	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
10	10	92	Checking	1800.5	533177005	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
11	11	3	Savings	1100.75	744264830	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
12	12	81	Checking	2700.0	4249714	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...
13	13	29	Savings	1300.25	3025925162	databricks	2025-05-06 20:3...	databricks	2025-05-06 20:30:...

Customers table:



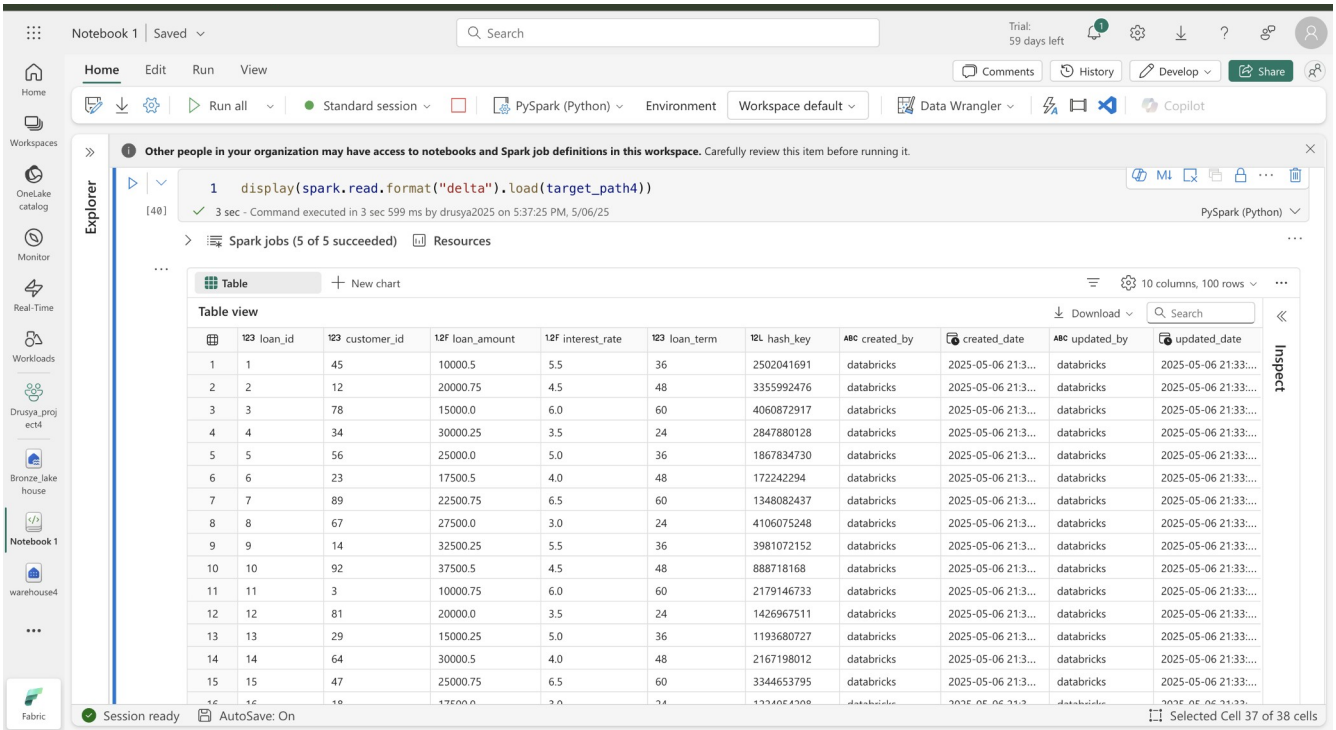
	customer_id	first_name	last_name	address	city	state	hash_key	created_by	created_date	updated_by	updated_dat
1	1	John	Doe	123 Elm St	Toronto	ON	2633702073	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
2	2	Jane	Smith	456 Maple A...	Ottawa	ON	822578646	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
3	3	Michael	Johnson	789 Oak Dr	Montreal	QC	3191181397	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
4	4	Emily	Davis	101 Pine Rd	Calgary	AB	494654746	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
5	5	David	Wilson	202 Birch Blvd	Vancouver	BC	1106639101	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
6	6	Emma	Clark	505 Cedar St	Halifax	NS	250108923	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
7	7	James	Martinez	606 Spruce Ln	Winnipeg	MB	1363658529	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
8	8	Olivia	Garcia	707 Fir St	Edmonton	AB	1254642380	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
9	9	William	Lopez	808 Redwoo...	Victoria	BC	3324949825	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
10	10	Ava	Anderson	909 Cypress ...	Quebec City	QC	49536236	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
11	11	Alexander	Thomas	1010 Willow...	St. John's	NL	1487697547	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
12	12	Isabella	Lee	1111 Poplar St	Fredericton	NB	363197713	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
13	13	Daniel	Harris	1212 Ash Blvd	Charlotteto...	PE	71082512	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
14	14	Sophia	Young	1313 Beech Dr	Yellowknife	NT	330418951	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::
15	15	Matthew	King	1414 Cedar Ln	Whitehorse	YT	2686390372	databricks	2025-05-06 21:2...	databricks	2025-05-06 21::

Loan payments table:



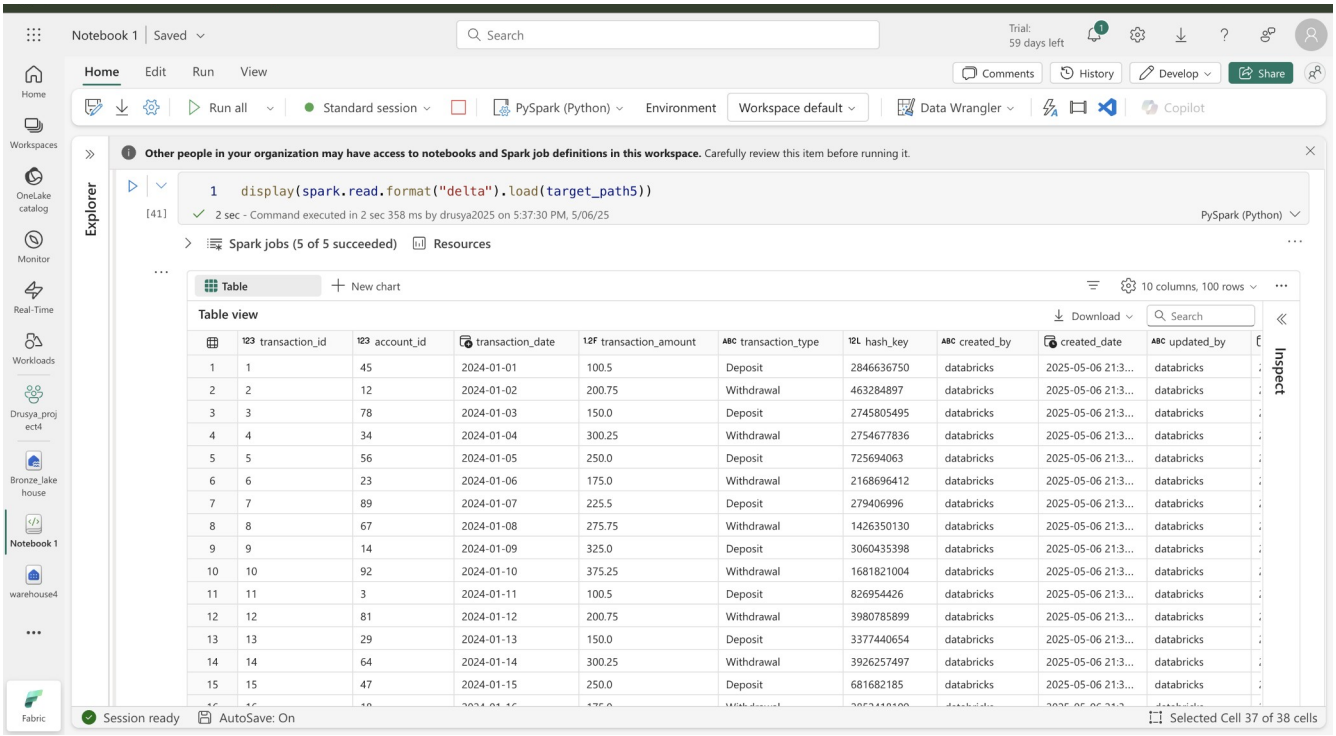
	payment_id	loan_id	payment_date	payment_amount	hash_key	created_by	created_date	updated_by	updated_date
1	1	45	2024-01-01	100.0	4189237233	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
2	2	23	2024-01-02	150.0	1457204963	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
3	3	67	2024-01-03	200.0	2777852429	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
4	4	89	2024-01-04	250.0	264335070	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
5	5	12	2024-01-05	300.0	438074744	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
6	6	34	2024-01-06	350.0	3330136997	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
7	7	56	2024-01-07	400.0	1192820904	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
8	8	78	2024-01-08	450.0	2203091019	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
9	9	90	2024-01-09	500.0	2681690964	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
10	10	11	2024-01-10	550.0	3325951132	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
11	11	22	2024-01-11	600.0	3894410316	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
12	12	33	2024-01-12	650.0	2585655648	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
13	13	44	2024-01-13	700.0	1774633368	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
14	14	55	2024-01-14	750.0	2348634532	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...
15	15	66	2024-01-15	800.0	1613741173	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:30:...

Loans table:



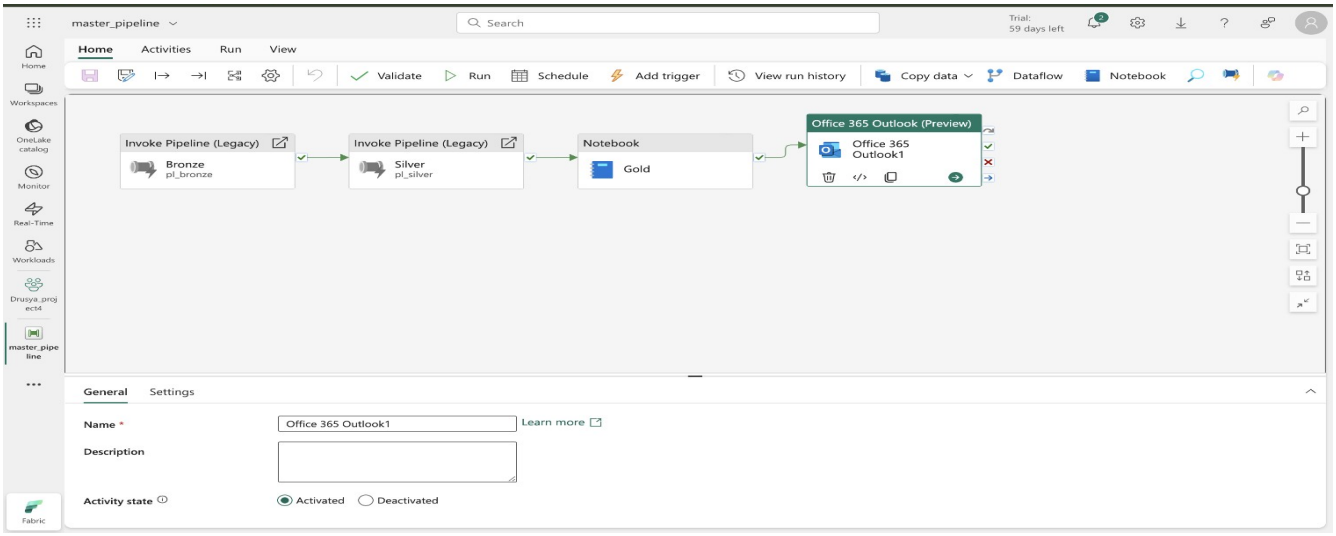
	loan_id	customer_id	loan_amount	interest_rate	loan_term	hash_key	created_by	created_date	updated_by	updated_date
1	1	45	10000.5	5.5	36	2502041691	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
2	2	12	20000.75	4.5	48	3355992476	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
3	3	78	15000.0	6.0	60	4060872917	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
4	4	34	30000.25	3.5	24	2847880128	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
5	5	56	25000.0	5.0	36	1867834730	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
6	6	23	17500.5	4.0	48	172242294	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
7	7	89	22500.75	6.5	60	1348082437	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
8	8	67	27500.0	3.0	24	4106075248	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
9	9	14	32500.25	5.5	36	3981072152	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
10	10	92	37500.5	4.5	48	888718168	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
11	11	3	10000.75	6.0	60	2179146733	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
12	12	81	20000.0	3.5	24	1426967511	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
13	13	29	15000.25	5.0	36	1193680727	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
14	14	64	30000.5	4.0	48	2167198012	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...
15	15	47	25000.75	6.5	60	3344653795	databricks	2025-05-06 21:3...	databricks	2025-05-06 21:33:...

Transactions table:

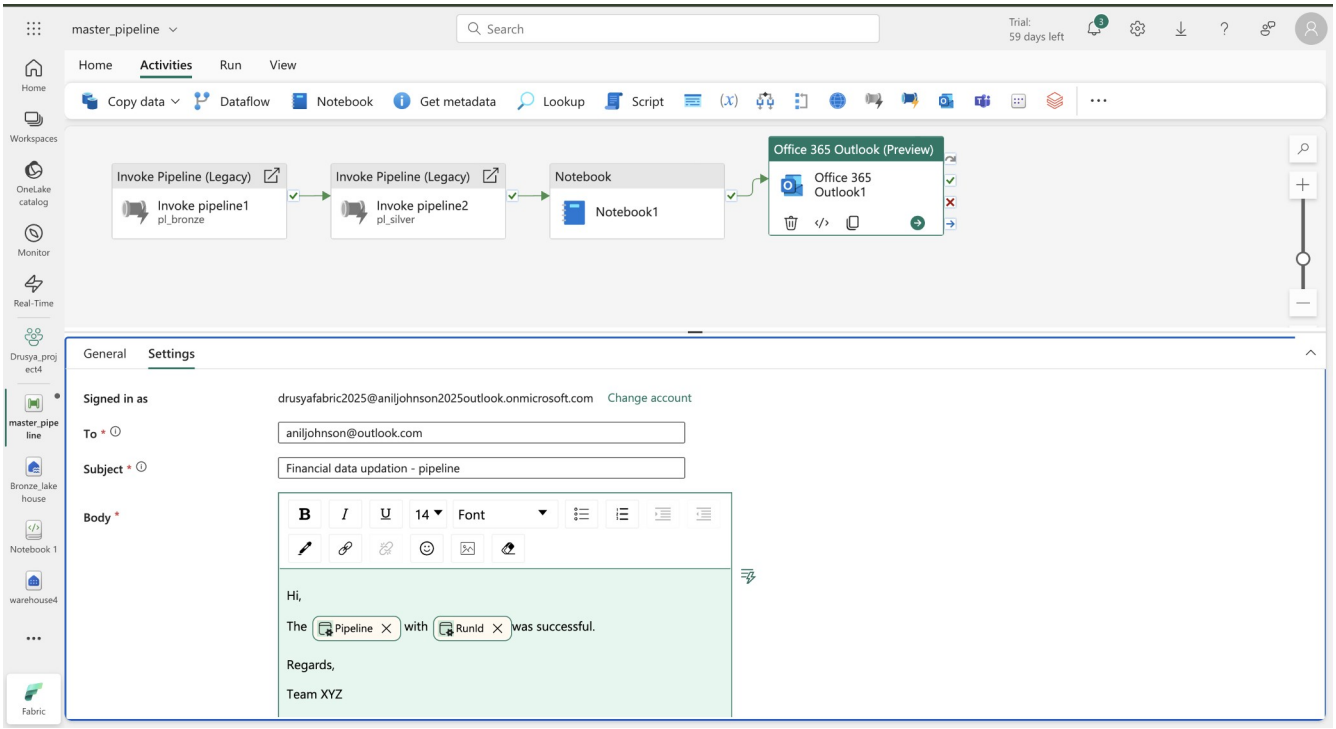


CREATING MASTER PIPELINE AND SCHEDULING THE PIPELINE WITH EMAIL NOTIFICATION TASK

Created a master pipeline and connected all the activities together.



Created an email activity so that we will get an email notification upon successful run of the pipeline. Since the email id I have is not supporting in Fabric it will show an error if we use an enterprise or company email it will work due to limitations I am only showing the demo of the email activity process.



Now I triggered my pipeline to run at specific interval of time.