

Dremio

Pedro González Fernández



16 de octubre de 2025

Índice

1. Instalando Dremio	2
2. Trabajando con formatos en Dremio	5
3. Transformando datos en Dremio	12

1. Instalando Dremio

En esta actividad vamos a instalar y aprender a usar la herramienta Dremio. Para ello vamos a comenzar con la configuración inicial:

Creamos un usuario de sistema y el grupo asociado:

```
pedro@bda:~$ sudo groupadd -r dremio
[sudo] contraseña para pedro:
pedro@bda:~$ sudo useradd -r -g dremio -d /var/lib/dremio -s /sbin/nologin dremio
pedro@bda:~$
```

```
pedro@bda:~$ cat /etc/group | grep dremio
dremio:x:985:
pedro@bda:~$ cat /etc/passwd | grep dremio
dremio:x:995:985::/var/lib/dremio:/sbin/nologin
pedro@bda:~$
```

También tenemos que preparar la estructura de directorios en /opt:

```
pedro@bda:~$ sudo mkdir /opt/dremio
pedro@bda:~$ sudo mkdir /opt/dremio/log
pedro@bda:~$ sudo mkdir /opt/dremio/run
pedro@bda:~$ sudo mkdir /opt/dremio/data
pedro@bda:~$ sudo chown -R dremio:dremio /opt/dremio
pedro@bda:~$ ls -lart /opt/dremio
total 20
drwxr-xr-x 7 root root 4096 oct 16 12:34 ..
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:34 log
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:34 run
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:34 data
drwxr-xr-x 5 dremio dremio 4096 oct 16 12:34 .
pedro@bda:~$
```

Ahora vamos a descargar el software de la [página oficial](#):

```
pedro@bda:~$ wget https://download.dremio.com/community-server/25.0.5-202407020141140611-dca7a083/dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz
2025-10-16 12:37:15-- https://download.dremio.com/community-server/25.0.5-202407020141140611-dca7a083/dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz
Resolviendo download.dremio.com (download.dremio.com)... 216.58.205.211, 2080:1450:4006:80e::2013
Conectando con download.dremio.com (download.dremio.com)[216.58.205.211]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/octet-stream]
Guardando como: 'dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz'
dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz [ 794.42M 13.8MB/s en 64s]
2025-10-16 12:38:19 (12.5 MB/s) - 'dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz' guardado [833087116]

pedro@bda:~$ ls
Descargas Documentos Imágenes Música Plantillas snap workspace
Desktop dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz kafka_2.13-3.0.0.tgz OnlineRetail.sql Publico Videos
pedro@bda:~$
```

Lo descomprimos en la ruta que hemos preparado:

```
pedro@bda:~$ sudo tar -xvzf dremio-community-25.0.5-202407020141140611-dca7a083.tar.gz -C /opt/dremio/ --strip-components=1
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-services-telemetry-tmpl-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-services-telemetry-api-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-services-telemetry-utils-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-dac-daemon-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-dac-utl-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-utl-lib-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-utl-common-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-js-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-hdfs-plugin-25.0.5-202407020141140611-dca7a083.jar
dremio-community-25.0.5-202407020141140611-dca7a083/jars/dremio-iceberg-plugin-25.0.5-202407020141140611-dca7a083.jar
```

```
pedro@bda:~$ sudo chown -R dremio:dremio /opt/dremio/
pedro@bda:~$ ls -lart /opt/dremio/
total 64
drwxr-xr-x 2 dremio dremio 4096 jul 2 2024 licenses
drwxr-xr-x 2 dremio dremio 4096 jul 2 2024 bin
drwxr-xr-x 7 root root 4096 oct 16 12:34 ..
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:34 log
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:34 run
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:34 data
drwxr-xr-x 3 dremio dremio 4096 oct 16 12:39 plugins
drwxr-xr-x 5 dremio dremio 20480 oct 16 12:39 jars
drwxr-xr-x 4 dremio dremio 4096 oct 16 12:39 lib
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:39 share
drwxr-xr-x 2 dremio dremio 4096 oct 16 12:39 conf
drwxr-xr-x 12 dremio dremio 4096 oct 16 12:39 .
```

Una vez hecho, vamos a crear un enlace simbólico de la configuración en /etc y copiamos el servicio para el arranque:

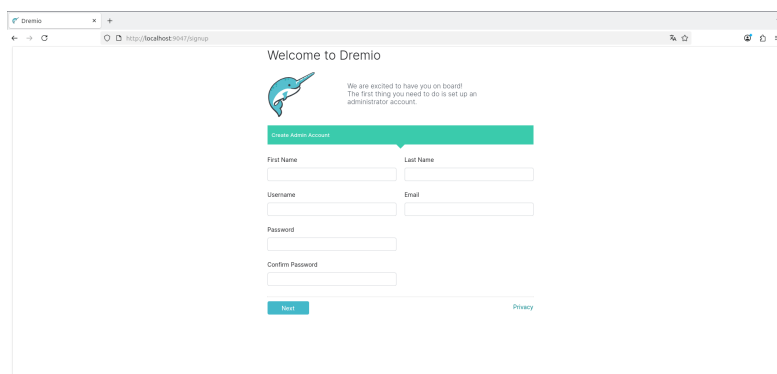
```
pedro@bda:~$ sudo ln -s /opt/dremio/conf /etc/dremio
pedro@bda:~$ sudo cp /opt/dremio/share/dremio.service /etc/systemd/system
```

Arrancamos dremio y, cuando finalice, lo configuramos para que se arranque al iniciar:

```
pedro@bda:~$ sudo systemctl daemon-reload
pedro@bda:~$ sudo systemctl start dremio
pedro@bda:~$ sudo systemctl status dremio
● dremio.service - Dremio Daemon Server
   Loaded: loaded (/etc/systemd/system/dremio.service; disabled; preset: enabled)
   Active: active (running) since Thu 2025-10-16 12:42:07 CEST; 30s ago
     Docs: https://docs.dremio.com
   Main PID: 2912 (java)
    Tasks: 51 (limit: 9365)
  Memory: 1.0G (peak: 1.0G)
     CPU: 52.871s
    CGroup: /system.slice/dremio.service
            └─2912 /usr/lib/jvm/java-11-openjdk-amd64/bin/java -Djava.util.logging.co
```

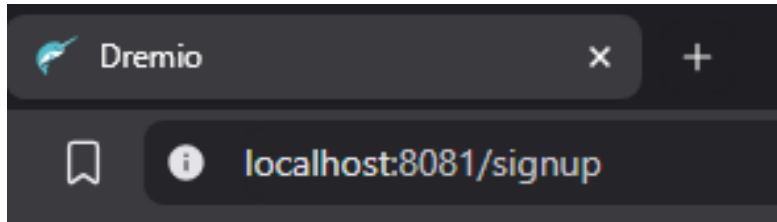
```
pedro@bda:~$ sudo systemctl enable dremio
Created symlink /etc/systemd/system/multi-user.target.wants/dremio.service → /etc/systemd/system/dremio.service.
pedro@bda:~$
```

Finalmente, vamos a comprobar si podemos acceder:



Como paso extra, voy a configurar un reenvío de puerto en la máquina virtual para acceder a dremio desde el host:

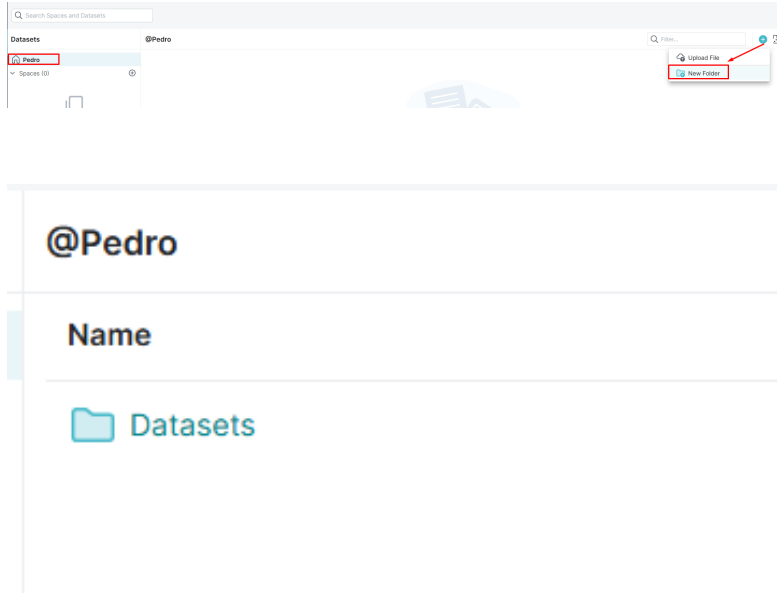
Nombre	Protocolo	IP anfitrión	Puerto anfitrión	IP invitado	Puerto invitado
Dremio	TCP		8081		9047
NiFi	TCP		8080		8081



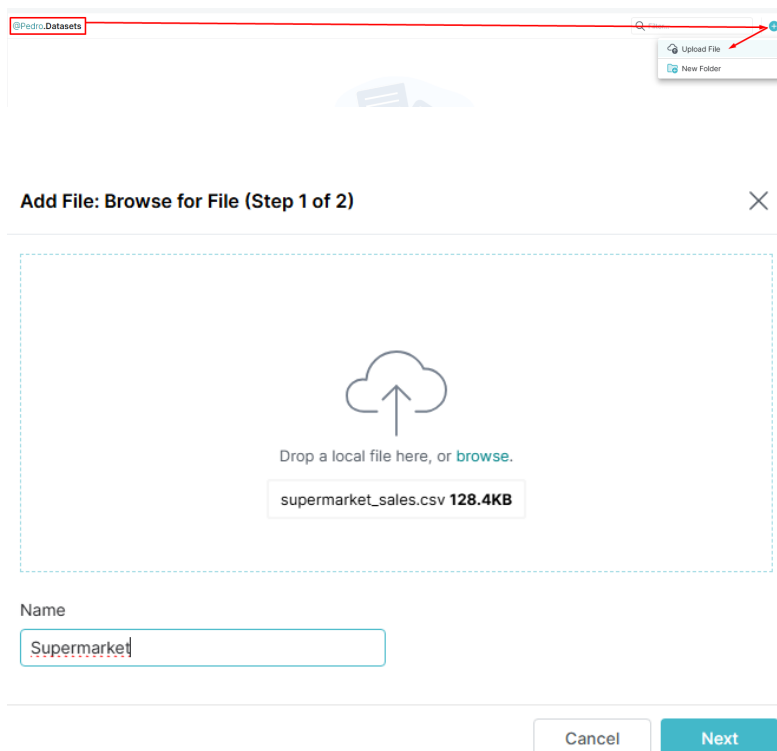
2. Trabajando con formatos en Dremio

En este apartado vamos a usar un dataset de muestra y hacer operaciones con él en diferentes formatos.

Primero creamos una carpeta en dremio para guardar todos los ficheros:



Dentro de la carpeta, tenemos que subir el archivo csv de muestra:



Al subirlo, tenemos que activar la opción para que extraiga los nombres de las columnas y no los trate como un registro más:

Add File: Set Format (Step 2 of 2)

Format
Text (delimited)

Column Delimiter: Comma Quote: Double Quote Comment: Number Sign #

Line Delimiter: CRLF - Windows Escape: \r\n Double Quote Options:

☒ Extract Column Names ☐ Skip First Line
☒ Trim Column Names

Invoice ID	Branch	City	Customer type	Gender	Product line
758-67-8428	A	Yangon	Member	Female	Health and beauty
226-31-3881	C	Naypyitaw	Normal	Female	Electronic accessories
631-41-3188	A	Yangon	Normal	Male	Home and lifestyle
123-19-1176	A	Yangon	Member	Male	Health and beauty
373-73-7918	A	Yangon	Normal	Male	Sports and travel
699-14-3826	C	Naypyitaw	Normal	Male	Electronic accessories
355-53-5943	A	Yangon	Member	Female	Electronic accessories
315-22-5665	C	Naypyitaw	Normal	Female	Home and lifestyle
665-32-9167	A	Yangon	Member	Female	Health and beauty
692-92-5582	B	Mandalay	Member	Female	Food and beverages
351-62-8822	B	Mandalay	Member	Female	Fashion accessories

Back Save

Datasets @Pedro.Datasets

Pedro

Spaces (0)

Supermarket

Si entramos en el dataset, podemos hacer consultas como si de una base de datos se tratase:

16 oct 2025, 15:44:10

Run Preview

Hide SQL pane Save as View

Context: (None selected)

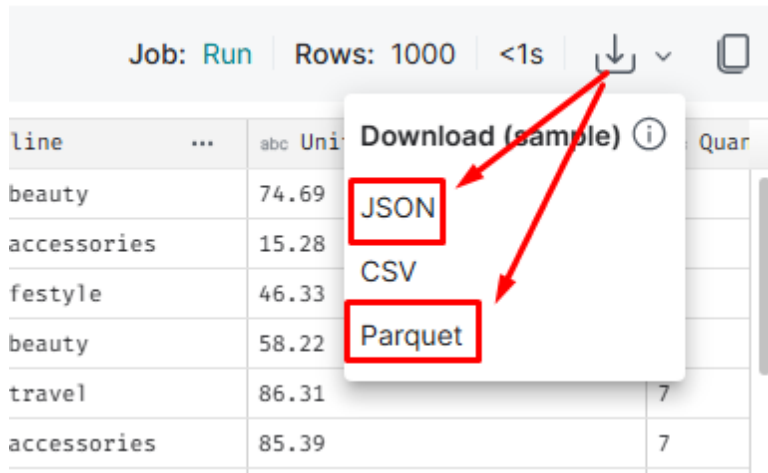
1 SELECT * FROM "Pedro".Datasets.Supermarket

Query1

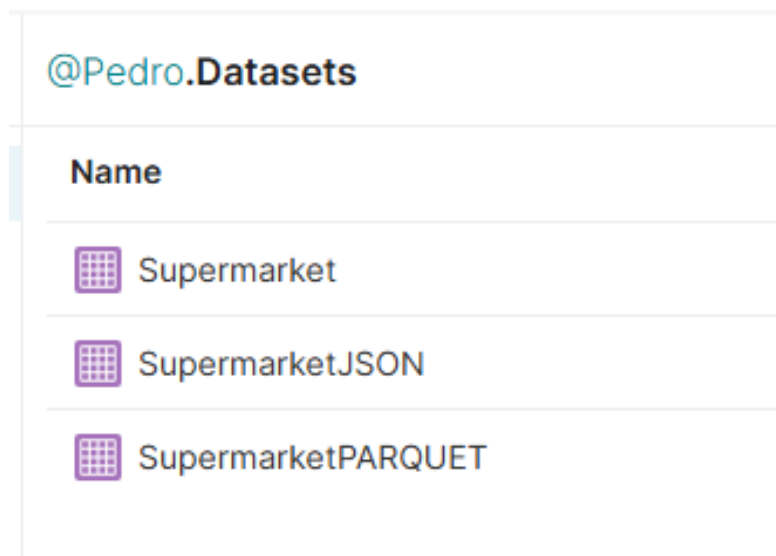
Add Column Group By Join Filter Columns 17 Columns Job: Run Rows: 1000 <1s

Invoice ID	Branch	City	Customer type	Gender	Product line
758-67-8428	A	Yangon	Member	Female	Health and beauty
226-31-3881	C	Naypyitaw	Normal	Female	Electronic accessories
631-41-3188	A	Yangon	Normal	Male	Home and lifestyle
123-19-1176	A	Yangon	Member	Male	Health and beauty
373-73-7918	A	Yangon	Normal	Male	Sports and travel
699-14-3826	C	Naypyitaw	Normal	Male	Electronic accessories
355-53-5943	A	Yangon	Member	Female	Electronic accessories
315-22-5665	C	Naypyitaw	Normal	Female	Home and lifestyle
665-32-9167	A	Yangon	Member	Female	Health and beauty
692-92-5582	B	Mandalay	Member	Female	Food and beverages

Hacemos una consulta para mostrar todos los datos y vamos a descargar este resultado en dos formatos, en JSON y en PARQUET:

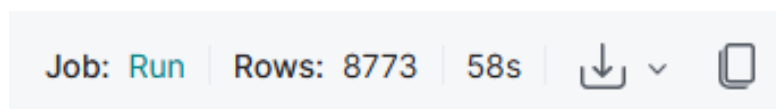


Y subimos los ficheros a Dremio como hicimos con el csv inicial:

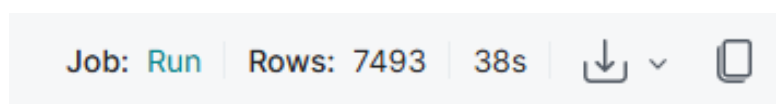


Ahora vamos a lanzar una consulta de prueba en cada uno de los datasets para comprobar que dataset la realiza más rápido:

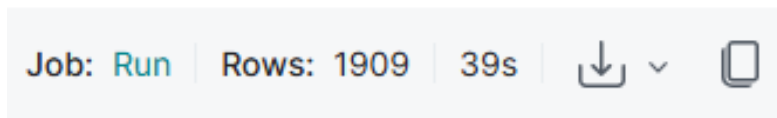
- CSV:



- JSON:



■ PARQUET:



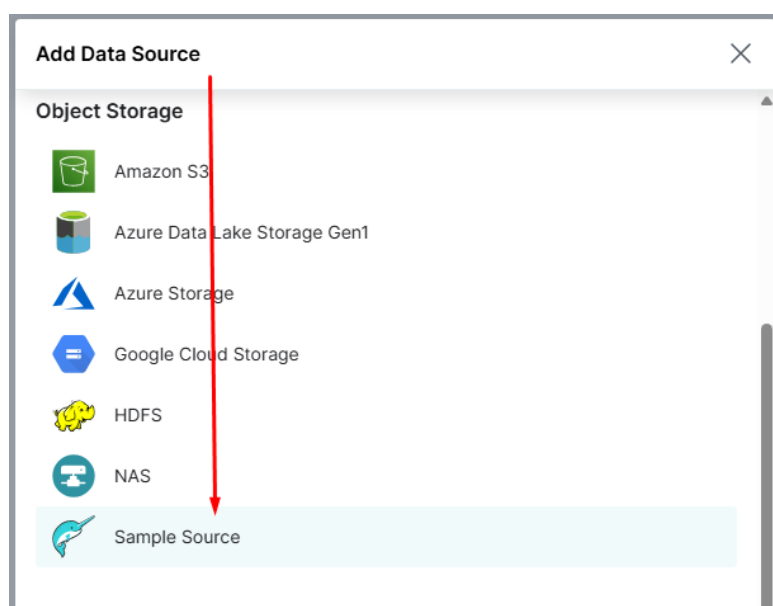
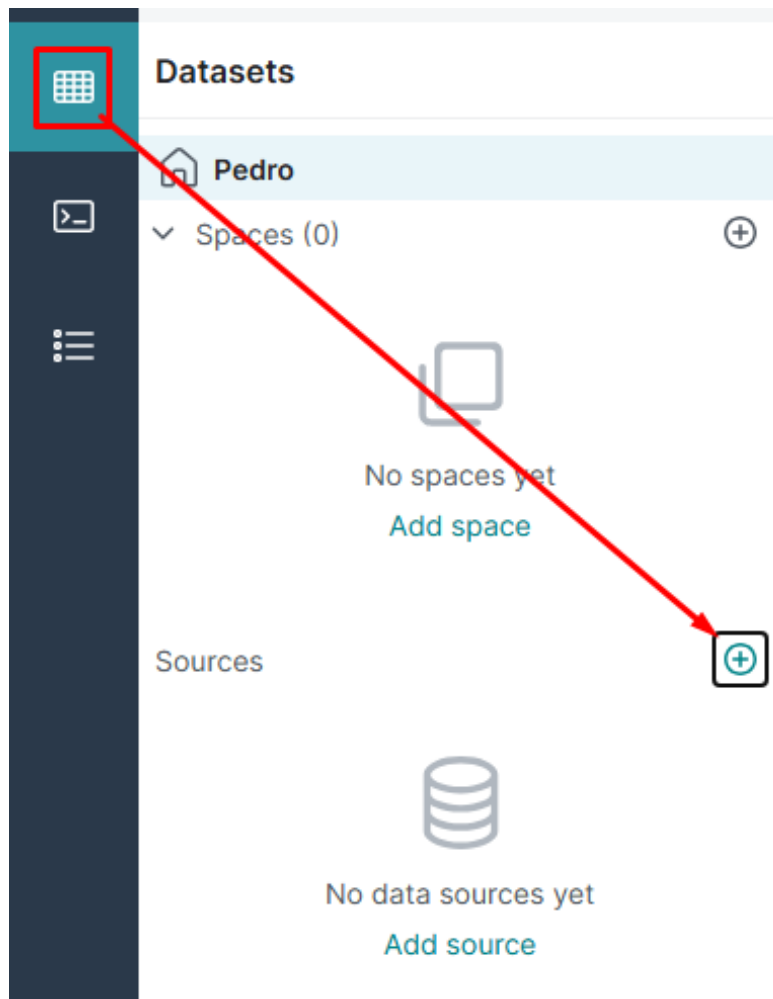
Podemos revisar los detalles de los trabajos:

Job ID	User	Dataset	Query Type	Queue	Start Time	Duration	SQL
...020-6410-1061-056a0c44100	Pedro	SupermarketPROJECT	UI (run)	LARGE	16/10/2025, 15:54:28	00:00:39.81	WITH CrossJoinData AS (SELECT f1."Invoice ID" AS InvoiceID, f1."City" ...
...08-0e3d-0a42-e3370dc0300	Pedro	SupermarketJSON	UI (run)	LARGE	16/10/2025, 15:53:32	00:00:38.81	WITH CrossJoinData AS (SELECT f1."Invoice ID" AS InvoiceID, f1."City" ...
...041-c420-4080-0504af62000	Pedro	Supermarket	UI (run)	LARGE	16/10/2025, 15:52:00	00:00:58.81	WITH CrossJoinData AS (SELECT f1."Invoice ID" AS InvoiceID, f1."City" ...
...f138-c224-4432-08549f63000	Pedro	Unavailable	UI (run)		16/10/2025, 15:50:30	<1s	WITH CrossJoinData AS (SELECT f1."Invoice ID" AS InvoiceID, f1."City" ...
...52b-748a-f0bc-cf502743400	Pedro	Supermarket	UI (download)	SMALL	16/10/2025, 15:47:50	00:00:02	CREATE TABLE "_datawarehouse"."75f70522-483a-4760-b0ad-76d57084617" S...

Summary

Status:	COMPLETED
Total Memory:	4.39 GB
CPU Used:	01m:52s
Query Type:	UI (run)
Start Time:	16/10/2025 15:54:28
Duration:	39.83s
Wait on Client:	<1s
User:	Pedro
Queue:	LARGE
Input:	283.27 KB / 6K Rows
Output:	253.40 KB / 1.9K Rows

Finalmente, vamos a realizar una comprobación. Queremos saber si PARQUET es autodescriptivo. Para esto vamos a cargar datos de prueba y crear un dataset a partir de un fichero parquet.



Datasets

Pedro

Spaces (0)

No spaces yet
Add space

Sources

Object Storage (1)

Samples

0











Samples

Name

samples.dremio.com

Dentro de los datos de prueba, hay varios ficheros en distinto formato:

Samples."samples.dremio.com"

Name
 Dremio University
 NYC-taxi-trips
 NYC-taxi-trips-iceberg
 NYC-taxi-trips.csv
 NYC-weather.csv
 SF weather 2018-2019.csv
 SF_incidents2016.json
 tpcds_sf1000
 zip_lookup.csv
 zips.json

Si miramos el csv, podemos ver que este no es autodescriptivo:

Line Delimiter: CRLF - Windows | Escape: \r\n | Double Quote: " | Options: ☐ Extract Column Names, ☐ Skip First Line, ☒ Trim Column Names

abc A	abc B	abc C	abc D	abc E
STATION	NAME	LATITUDE	LONGITUDE	ELEVATION
USW00023272	SAN FRANCISCO DOWNTOWN, CA US	37.7705	-122.4269	45.7
USW00023272	SAN FRANCISCO DOWNTOWN, CA US	37.7705	-122.4269	45.7
USW00023272	SAN FRANCISCO DOWNTOWN, CA US	37.7705	-122.4269	45.7
USW00023272	SAN FRANCISCO DOWNTOWN, CA US	37.7705	-122.4269	45.7
USW00023272	SAN FRANCISCO DOWNTOWN, CA US	37.7705	-122.4269	45.7

Sin embargo, si miramos el JSON, veremos que si lo es:

Format: JSON

abc IncidentNum	abc Category	abc Descript	abc DayOfWeek	abc Date
120058272	WEAPON LAWS	POSS OF PROHIBITED WEAPON	Friday	2016-01-29
120058272	WEAPON LAWS	FIREARM, LOADED, IN VEHICLE,	Friday	2016-01-29
141059263	WARRANTS	WARRANT ARREST	Monday	2016-04-25
160013662	NON-CRIMINAL	LOST PROPERTY	Tuesday	2016-01-05
160002740	NON-CRIMINAL	LOST PROPERTY	Friday	2016-01-01
160002869	ASSAULT	BATTERY	Friday	2016-01-01
160003130	OTHER OFFENSES	PAROLE VIOLATION	Saturday	2016-01-02
160003259	NON-CRIMINAL	FIRE REPORT	Saturday	2016-01-02

Vamos a comprobar que ocurre con un archivo PARQUET.

Samples.samples.dremio.com.Dremio University

Name

- 100_Sales_Records_inconsistency.csv
- 4week_recipes.json
- aac_shelter_outcomes.csv
- airbnb_listings.csv
- employees.parquet**
- googleplaystore.csv

Format: Parquet

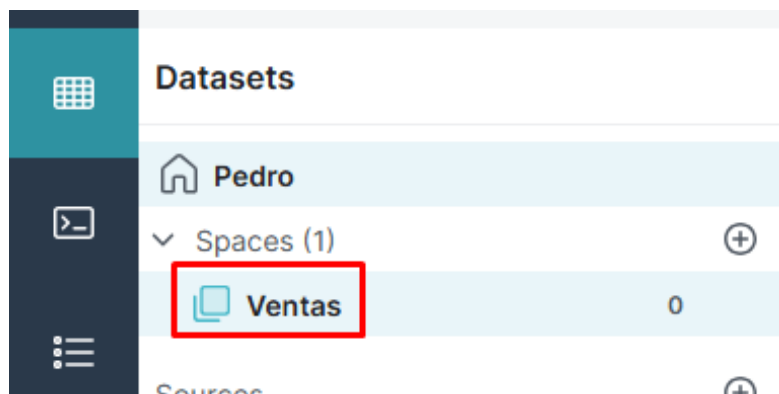
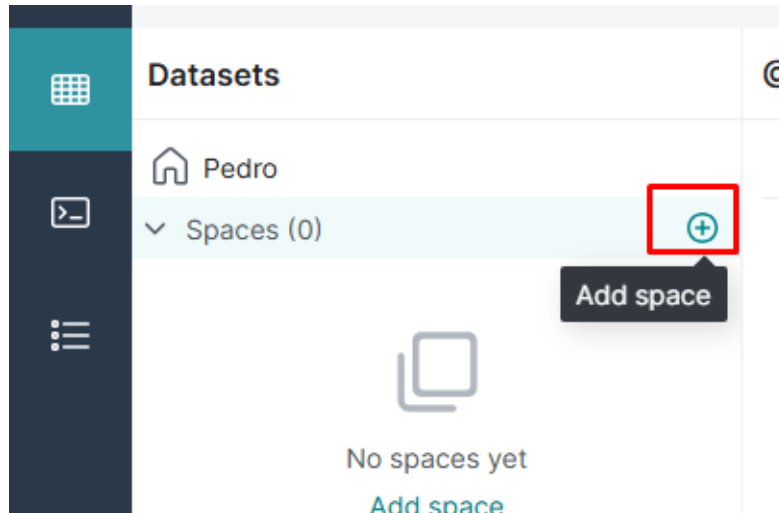
## employee_id	abc first_name	abc last_name	abc email	abc phone_number
100.0	Steven	King	SKING	515.123.4567
101.0	Neena	Kochhar	NKOCHHAR	515.123.4568
102.0	Lex	De Haan	LDEHAAN	515.123.4569
103.0	Alexander	Hunold	AHUNOLD	590.423.4567
104.0	Bruce	Ernst	BERNST	590.423.4568
105.0	David	Austin	DAUSTIN	590.423.4569
106.0	Valli	Pataballa	VPATABAL	590.423.4568
107.0	Diana	Lorentz	DLARENTZ	590.423.5567
108.0	Nancy	Greenberg	NGREENBE	515.124.4569

Como podemos observar, PARQUET si es autodescriptivo.

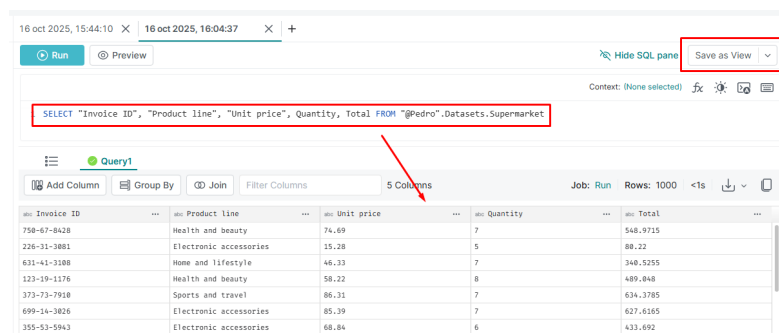
3. Transformando datos en Dremio

En este apartado vamos a empezar a partir de un dataset y vamos a transformar los datos a nuestra elección.

Para ello, vamos a crear un espacio llamado 'Ventas':



Ahora vamos a realizar una consulta a la tabla Supermarket y guardamos el resultado como un dataset nuevo dentro del espacio que hemos creado:



Lo guardamos como 'Ventas':

Save View As

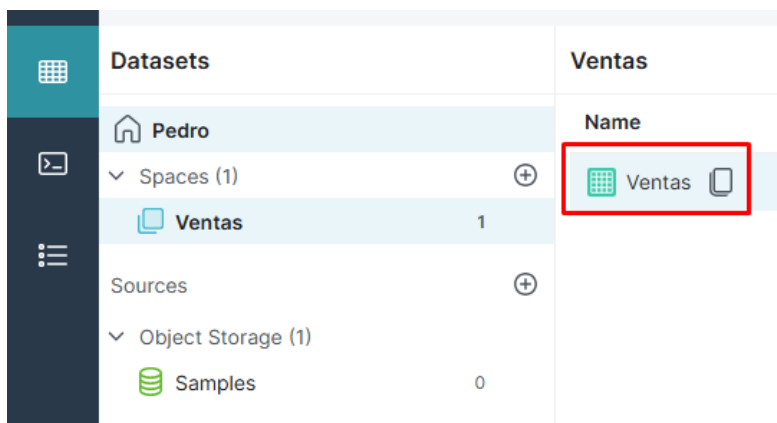
Name

Ventas

Location

> @Pedro



▼  Ventas



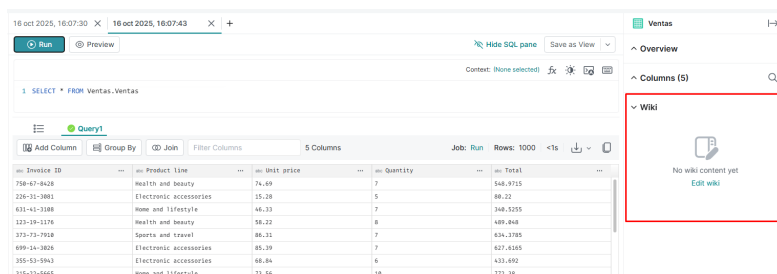
Datasets

Ventas

Name

 Ventas 

Lo primero que vamos a hacer es crear una wiki del dataset. Esto nos permite tener la documentación centralizada, aportando contexto y trazabilidad a los datos.



16 oct 2025, 16:07:30 X 16 oct 2025, 16:07:43 X +

Run Preview Hide SQL pane Save as View

Context: (None selected) SQL Query Editor

1 SELECT * FROM Ventas.Ventas

Query1

Add Column Group By Join Filter Columns 5 Columns Job: Run Rows: 1000 <1s

Invoice ID	Product Line	Unit Price	Quantity	Total
758-67-8428	Health and beauty	74.49	7	548.9715
228-21-3881	Electronic accessories	15.28	5	88.22
631-41-1188	Home and lifestyle	44.33	7	248.3255
133-18-1178	Health and beauty	18.22	8	169.848
373-73-7918	Sports and travel	88.21	7	634.3785
699-1A-3826	Electronic accessories	85.39	7	627.8165
355-53-59A3	Electronic accessories	68.84	6	433.492
116-73-6245	Home and lifestyle	71.64	18	779.18

Ventas

Overview

Columns (5)

Wiki

No wiki content yet [Edit wiki](#)

Añadimos una descripción al dataset:

Wiki

Descripción General:
Este dataset contiene información relacionada con transacciones de facturación. Cada registro representa una transacción con detalles sobre la línea de productos, precios unitarios, cantidad y el total de la factura. Los campos principales y sus descripciones son los siguientes:

Campos:

****Invoice ID:****
Tipo de dato: Texto/Numérico.
Descripción: Identificador único de cada factura. Este campo asegura que cada transacción de facturación pueda ser rastreada individualmente.
Ejemplo: 750-67-8428.

****Product Line:****
Tipo de dato: Texto.
Descripción: Categoría o línea de productos asociados a la transacción. Indica el tipo de producto o servicio que se incluye en la factura.
Ejemplo: "Health and beauty".

****Unit Price:****
Tipo de dato: Numérico (Decimal).
Descripción: Precio unitario de un producto o servicio en la transacción, antes de aplicar cualquier descuento o impuesto.
Ejemplo: 19.99 (indica que cada unidad cuesta 19.99 de la moneda correspondiente).

****Quantity:****
Tipo de dato: Numérico (Entero).
Descripción: Cantidad de unidades del producto o servicio facturado. Este campo multiplica el precio unitario para calcular el subtotal.
Ejemplo: 3 (tres unidades).

Cancel Save

Y ahora, cuando hagamos consultas, podemos tener la documentación siempre a mano:

16 oct 2025, 16:07:30 X
16 oct 2025, 16:07:43 X +

Run Preview
Hide SQL pane Save as View

Context: (None selected)

1 SELECT * FROM Ventas.Ventas

Query1

Add Column Group By Join Filter Columns 5 Columns Job: Run Rows: 1000 <1s

Invoice ID	Product Line	Unit price	Quantity	Total
750-67-8428	Health and beauty	14.69	7	548.9715
226-31-3881	Electronic accessories	15.28	5	88.22
631-41-3188	Home and lifestyle	46.33	7	244.3255
121-19-1176	Health and beauty	58.22	8	465.848
273-75-7928	Sports and travel	86.10	7	622.705
699-14-3826	Electronic accessories	85.39	7	627.635
355-53-5943	Electronic accessories	68.84	6	433.092
315-22-5665	Home and lifestyle	73.56	10	735.56
665-32-9167	Health and beauty	36.28	2	72.56
607-20-5582	Food and beverages	54.26	3	162.78
351-42-8822	Fashion accessories	14.48	4	57.92
529-56-3974	Electronic accessories	25.51	4	102.04
365-64-8515	Electronic accessories	46.95	5	234.75
252-56-2899	Food and beverages	43.19	10	431.9
609-36-3918	Health and beauty	71.18	10	711.8
299-48-1885	Sports and travel	63.72	6	382.32
656-95-9349	Health and beauty	68.93	7	482.51
765-26-6951	Sports and travel	72.41	6	434.46
329-42-1586	Food and beverages	54.67	3	164.01
319-88-3148	Home and lifestyle	48.3	2	96.6
388-71-4485	Electronic accessories	66.86	5	334.3

Ventas

Overview

Columns (5)

Wiki

Descripción General:
Este dataset contiene información relacionada con transacciones de facturación. Cada registro representa una transacción con detalles sobre la línea de productos, precios unitarios, cantidad y el total de la factura. Los campos principales y sus descripciones son los siguientes:

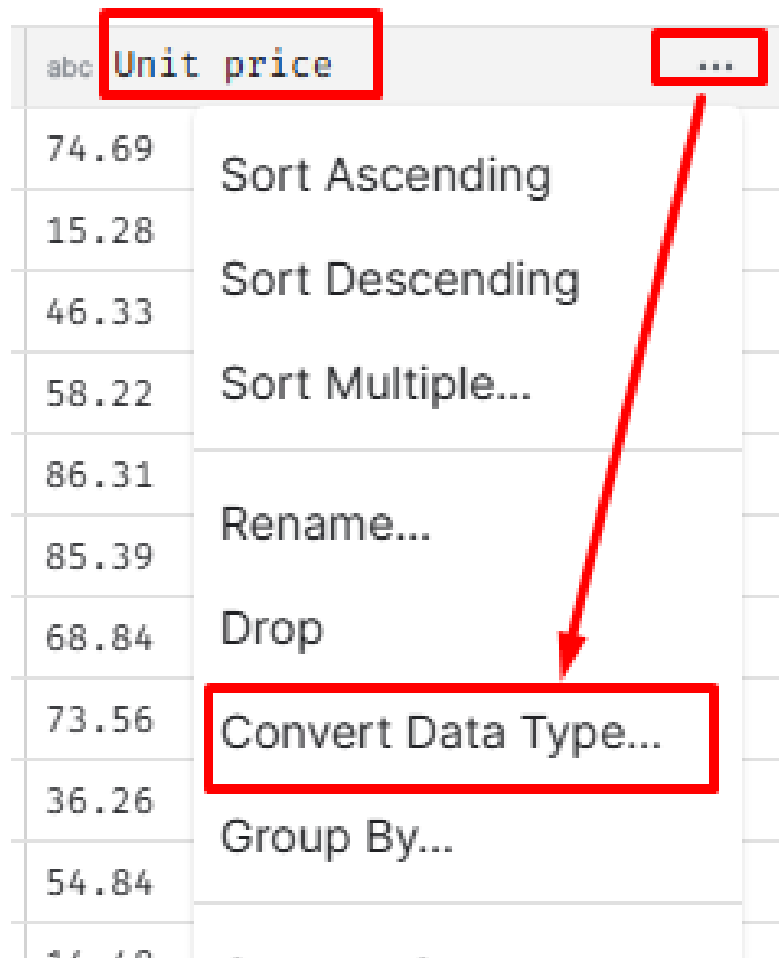
Campos:

Invoice ID:
Tipo de dato: Texto/Numérico.
Descripción: Identificador único de cada factura. Este campo asegura que cada transacción de facturación pueda ser rastreada individualmente.
Ejemplo: 750-67-8428.

Product Line:
Tipo de dato: Texto.
Descripción: Categoría o línea de productos asociados a la transacción. Indica el tipo de producto o servicio que se incluye en la factura.
Ejemplo: "Health and beauty".

Unit Price:
Tipo de dato: Numérico (Decimal).

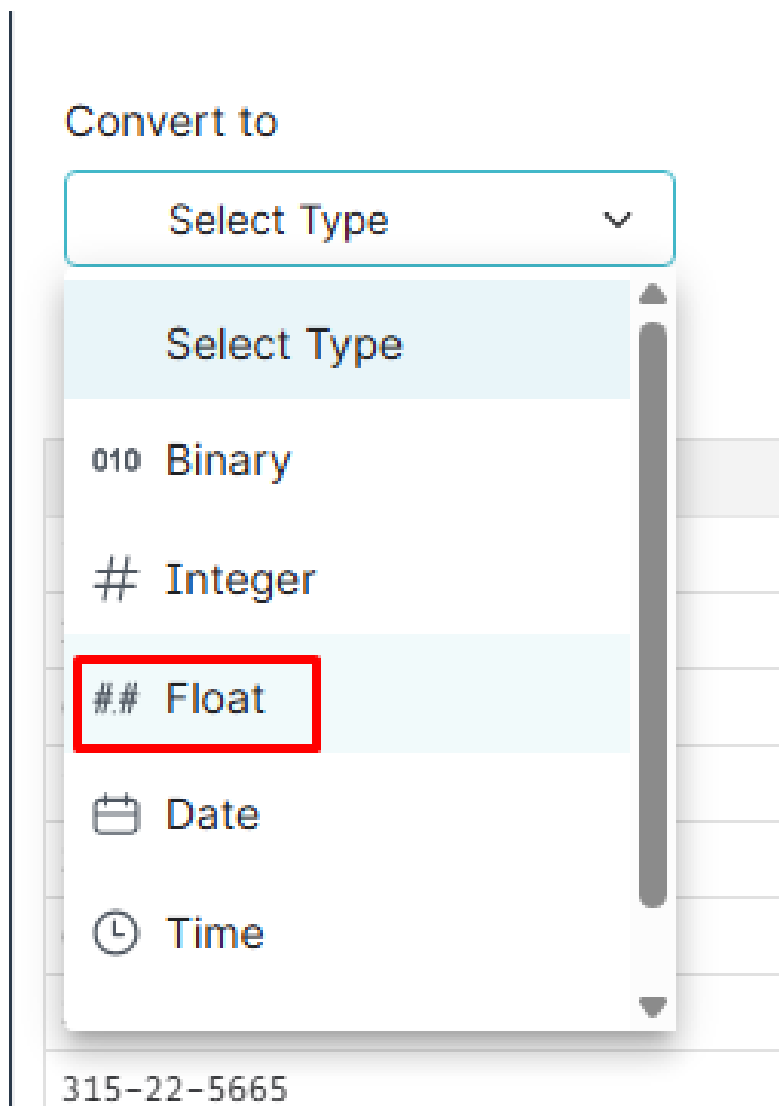
Lo siguiente que vamos a hacer es cambiar los tipos de datos. Empezaremos por cambiar a formato número (float o entero) los campos numéricos pero que están puestos en formato texto. Seleccionamos el campo y le damos a 'Convert data Type...'



The image shows a data table with a header row and several data rows. The header row has a column labeled 'Unit price'. A context menu is open over the 'Unit price' column, showing options: 'Sort Ascending', 'Sort Descending', 'Sort Multiple...', 'Rename...', 'Drop', 'Convert Data Type...', and 'Group By...'. A red arrow points from the '...' button in the header to the 'Convert Data Type...' option in the menu. Red boxes highlight the 'Unit price' header and the 'Convert Data Type...' option.

abc	Unit price	...
	74.69	Sort Ascending
	15.28	Sort Descending
	46.33	Sort Multiple...
	58.22	Rename...
	86.31	Drop
	85.39	Convert Data Type...
	68.84	Group By...
	73.56	
	36.26	
	54.84	
	41.10	

Dentro, ponemos el tipo de campo al que queremos convertirlo:



Y vemos un preview del resultado:

Search Options and Defaults

16 Oct 2025, 16:07:43

Change Data Type

Convert to: **## Float**

Action for Non-matching values:

- ☒ Replace values with null
- ☐ Replace values with:

Delete Rows

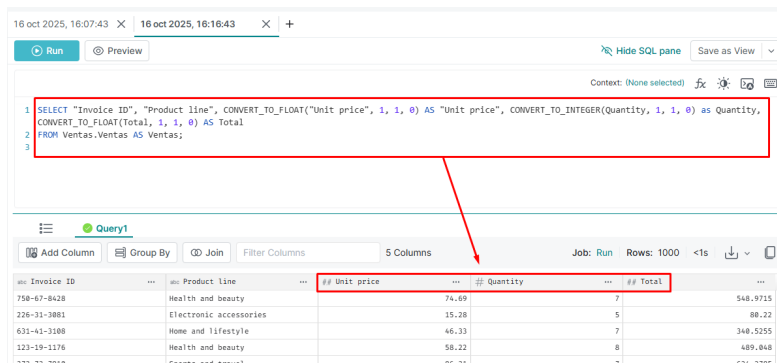
New Field Name: Options: ☒ Drop Source Field (and price)

Unit price ☒ Drop Source Field (and price)

Apply Preview Cancel Result based on sample dataset

Product ID	Product Line	Unit price	Unit price (new)	Quantity	Total
700-01-0028	Health and beauty	74.89	74.89	7	524.23
100-10-1001	Electronic accessories	17.39	17.39	5	86.95
510-10-1008	Home and lifestyle	46.33	46.33	7	324.31
100-10-1076	Health and beauty	58.22	58.22	8	465.76
100-10-1088	Health and beauty	86.36	86.36	7	604.52
400-10-1008	Electronic accessories	89.39	89.39	7	625.73
100-10-1003	Electronic accessories	66.86	66.86	8	534.88
100-10-1003	Home and lifestyle	77.39	77.39	28	2166.52
400-10-1007	Health and beauty	36.26	36.26	2	72.52
400-10-1002	Food and beverages	54.86	54.86	3	164.58
100-10-1002	Electronic accessories	19.48	19.48	8	155.84
400-10-1003	Electronic accessories	25.51	25.51	8	204.08

Si aplicamos, vemos la consulta generada automáticamente para la conversión del tipo. Si repetimos el proceso con el resto de campos, llegamos a este resultado:




The screenshot shows a SQL query editor with a query that converts 'Unit price' to a float and 'Quantity' to an integer. Below the query, a preview table is displayed with columns: Invoice ID, Product line, Unit price, Quantity, and Total. The table contains data for various product lines like Health and beauty, Electronic accessories, Home and lifestyle, and Health and beauty.

```
1 SELECT "Invoice ID", "Product line", CONVERT_TO_FLOAT("Unit price", 1, 1, 0) AS "Unit price", CONVERT_TO_INTEGER("Quantity", 1, 1, 0) AS "Quantity",  
2 CONVERT_TO_FLOAT("Total", 1, 1, 0) AS "Total"  
3 FROM Ventas.Ventas AS Ventas;
```

Invoice ID	Product line	Unit price	Quantity	Total
750-67-0428	Health and beauty	74.69	7	548.9715
226-31-3881	Electronic accessories	15.28	5	88.22
631-41-3108	Home and lifestyle	46.33	7	340.5255
123-19-1176	Health and beauty	58.22	8	489.848

Ahora vamos a limpiar los espacios de las cadenas y capitalizar el texto:



The screenshot shows a context menu for a table column named 'Product line'. The menu includes options like 'Sort Ascending', 'Sort Descending', 'Sort Multiple...', 'Rename...', 'Drop', 'Convert Data Type...', 'Group By...', 'Convert Case...', 'Trim Whitespace...', and 'Calculated Field...'. The 'Convert Case...' and 'Trim Whitespace...' options are highlighted with a red box.

abc Product line
Sports
Home ar
Fashion
Fashion
Food ar
Health
Fashion
Sports
Sports
Health
Food ar
Sports
Sports
Electro
Health

- Sort Ascending
- Sort Descending
- Sort Multiple...
- Rename...
- Drop
- Convert Data Type...
- Group By...
- Convert Case...
- Trim Whitespace...
- Calculated Field...

16 oct 2025, 16:16:43

Convert Case

Options

☐ UPPERCASE

☐ lowercase

☒ Title Case

New Field Name

Options

Product line

☒ Drop Source Field (Product line)

Apply

Preview

Cancel

Result based on sample dataset

abc Invoice ID	abc Product Line	abc Product line (new)
549-59-1358	Sports and travel	Sports And Travel
227-83-5010	Home and lifestyle	Home And Lifestyle
649-29-6775	Fashion accessories	Fashion Accessories
189-17-4241	Fashion accessories	Fashion Accessories
145-94-9061	Food and beverages	Food And Beverages
848-62-7243	Health and beauty	Health And Beauty
871-79-8483	Fashion accessories	Fashion Accessories
149-71-6266	Sports and travel	Sports And Travel
640-49-2076	Sports and travel	Sports And Travel
595-11-5460	Health and beauty	Health And Beauty
183-56-6882	Food and beverages	Food And Beverages
232-16-2483	Sports and travel	Sports And Travel
129-29-8530	Sports and travel	Sports And Travel
272-65-1806	Electronic accessories	Electronic Accessories
333-73-7901	Health and beauty	Health And Beauty
777-82-7220	Home and lifestyle	Home And Lifestyle

16 oct 2025, 16:16:43

Trim Whitespace

Options

☒ Trim from both sides

☐ Trim from the start (Trim Left)

☐ Trim from the end (Trim Right)

New Field Name

Options

Product line

☒ Drop Source Field (Product line)

Apply

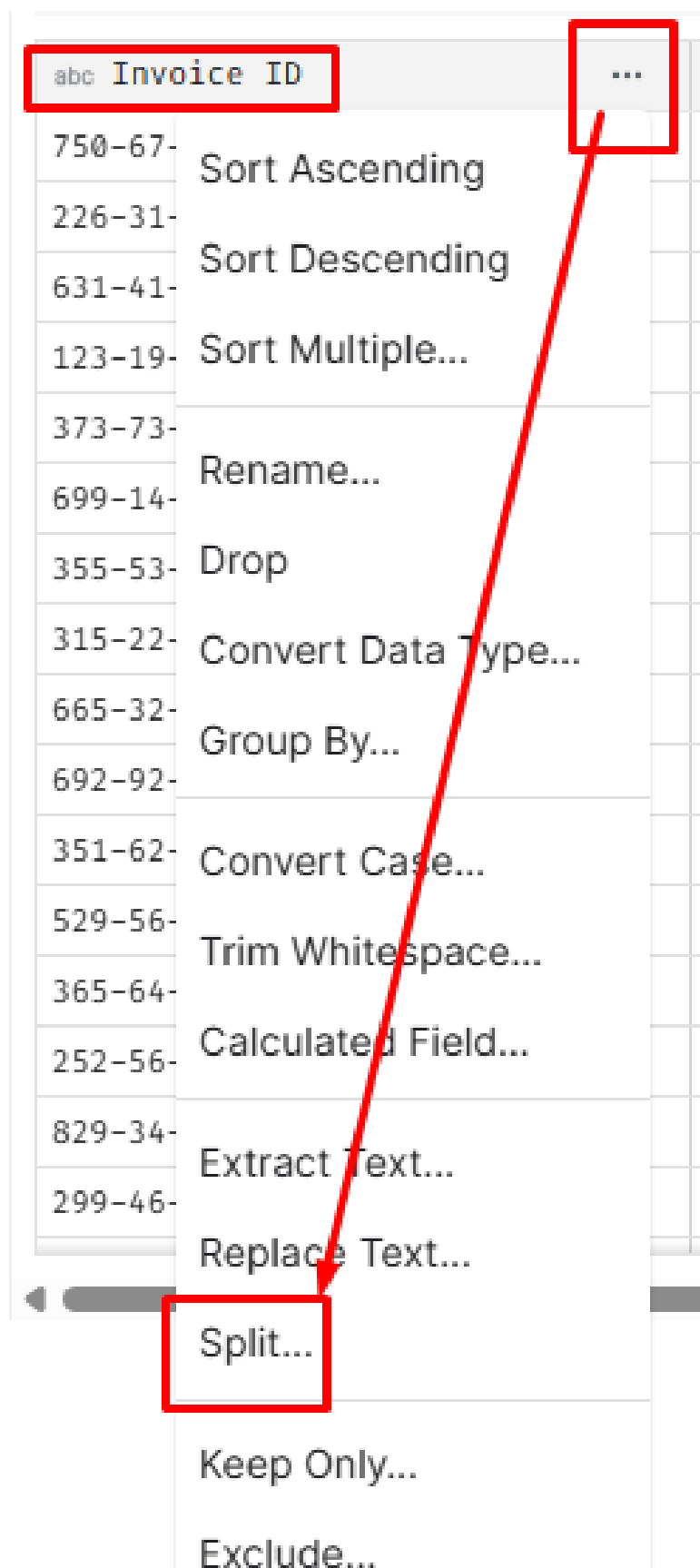
Preview

Cancel

Result based on sample dataset

abc Invoice ID	abc Product Line	abc Product line (new)
750-67-8428	Health And Beauty	Health And Beauty
226-31-3081	Electronic Accessories	Electronic Accessories
631-41-3108	Home And Lifestyle	Home And Lifestyle
123-19-1176	Health And Beauty	Health And Beauty
373-73-7910	Sports And Travel	Sports And Travel
699-14-3026	Electronic Accessories	Electronic Accessories
355-53-5943	Electronic Accessories	Electronic Accessories
315-22-5665	Home And Lifestyle	Home And Lifestyle
665-32-9167	Health And Beauty	Health And Beauty
692-92-5582	Food And Beverages	Food And Beverages
351-62-0822	Fashion Accessories	Fashion Accessories
529-56-3974	Electronic Accessories	Electronic Accessories
365-64-0515	Electronic Accessories	Electronic Accessories
252-56-2699	Food And Beverages	Food And Beverages

Y, finalmente, vamos a separar por partes el ID y luego lo vamos a volver a unir:



16 oct 2025, 16:16:43

Replace Extract Split Keep Only Exclude

Edit Selection

Fixed String -

Ignore Case

1000 matched values 0 unmatched values

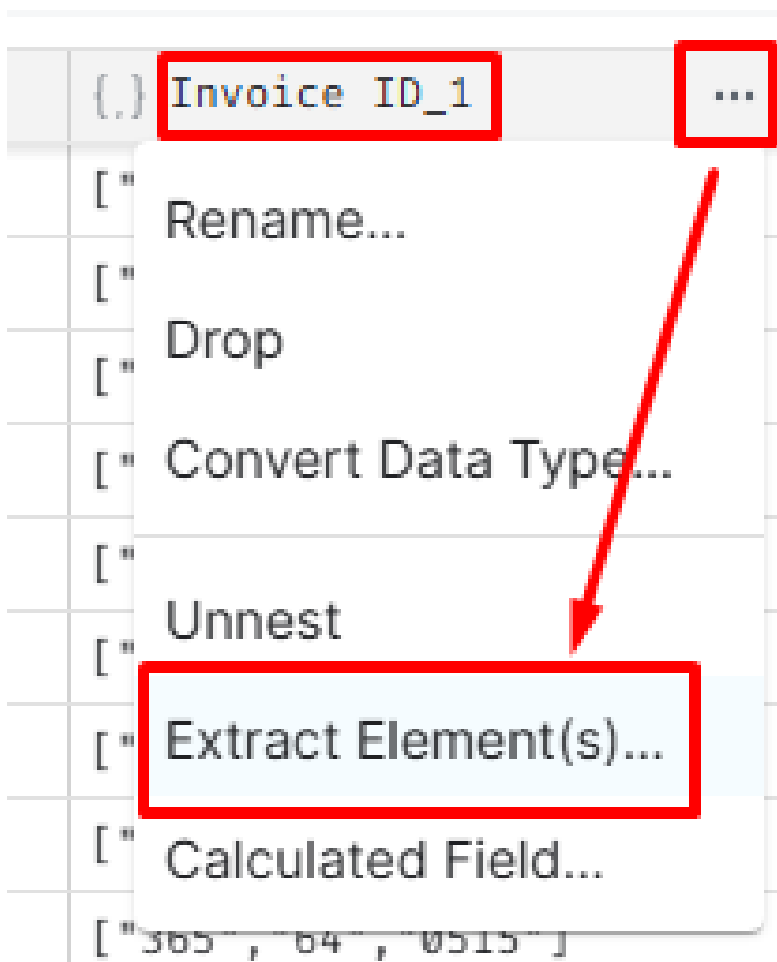
or, select text in the field to see more patterns

Position: All Max Fields: 10 New Field Name: Invoice ID_1 Options: ☐ Drop Source Field (Invoice ID)

Apply Preview Cancel Result based on sample dataset

Invoice ID	Invoice ID_1	Product line
750-67-8428	["750","67","8428"]	Health And Beauty
226-31-3081	["226","31","3081"]	Electronic Accessories
631-41-3108	["631","41","3108"]	Home And Lifestyle
123-19-1176	["123","19","1176"]	Health And Beauty
373-73-7910	["373","73","7910"]	Sports And Travel
699-14-3026	["699","14","3026"]	Electronic Accessories
355-53-5943	["355","53","5943"]	Electronic Accessories
315-22-5665	["315","22","5665"]	Home And Lifestyle
665-32-9167	["665","32","9167"]	Health And Beauty
692-92-5582	["692","92","5582"]	Food And Beverages
351-62-0822	["351","62","0822"]	Fashion Accessories

* Aquí es importante no borrar el campo original.



16 oct 2025, 16:16:43

Replace **Extract** Split Keep Only Exclude

Extract ☒ Single ☐ Multiple

Element
0

1000 matched values 0 unmatched values

New Field Name
Invoice ID_1

Options
☐ Drop Source Field (Invoice ID_1)

Apply Preview Cancel ⚠ Result based on sample dataset

abc Invoice ID	(.) Invoice ID_1	abc Invoice ID_1_1
750-67-8428	["750","67","8428"]	750
226-31-3081	["226","31","3081"]	226
631-41-3108	["631","41","3108"]	631
123-19-1176	["123","19","1176"]	123
373-73-7910	["373","73","7910"]	373
699-14-3026	["699","14","3026"]	699
355-53-5943	["355","53","5943"]	355
315-22-5665	["315","22","5665"]	315
665-32-9167	["665","32","9167"]	665
692-92-5582	["692","92","5582"]	692
351-62-0822	["351","62","0822"]	351
529-56-3974	["529","56","3974"]	529

{. Invoice ID_1 ...}

- Rename...
- Drop
- Convert Data Type...
- Unnest
- Extract Element(s)...
- Calculated Field...

16 oct 2025, 16:16:43

Add Calculated Field

fx🔍📄🗑️

1CONCAT("Invoice ID_1"[0], '_', "Invoice ID_1"[1], '_', "Invoice ID_1"[2])

New Field Name

Invoice ID_1_2

Options

☒ Drop Source Field (Invoice ID_1)

ApplyPreviewCancel

⚠️ Result based on sample dataset

Invoice ID	Invoice ID_1	Invoice ID_1_2	Invoice ID_1_1
750-67-8428	["750","67","8428"]	750_67_8428	750
226-31-3081	["226","31","3081"]	226_31_3081	226
631-41-3108	["631","41","3108"]	631_41_3108	631
123-19-1176	["123","19","1176"]	123_19_1176	123
373-73-7910	["373","73","7910"]	373_73_7910	373
699-14-3026	["699","14","3026"]	699_14_3026	699
355-53-5943	["355","53","5943"]	355_53_5943	355
315-22-5665	["315","22","5665"]	315_22_5665	315
665-32-9167	["665","32","9167"]	665_32_9167	665
692-92-5582	["692","92","5582"]	692_92_5582	692
351-62-0822	["351","62","0822"]	351_62_0822	351
529-56-3974	["529","56","3974"]	529_56_3974	529
365-64-0515	["365","64","0515"]	365_64_0515	365
252-56-2699	["252","56","2699"]	252_56_2699	252

La consulta final quedaría así:

```
SELECT "Invoice ID", CONCAT("Invoice ID_1"[0], '_', "Invoice ID_1"[1], '_', "Invoice ID_1"[2]) AS "Invoice ID_1_2", "Invoice ID_1_1", "Product line", "Unit price", Quantity, Total
FROM (
  SELECT "Invoice ID", "Invoice ID_1", "Invoice ID_1_2" AS "Invoice ID_1_1", "Product line", "Unit price", Quantity, Total
  FROM (
    SELECT "Invoice ID", regexp_split("Invoice ID", '[0-9-]', 'ALL', 10) AS "Invoice ID_1", trim(both ' ' from TITLE("Product line")) AS "Product line", CONVERT_TO_FLOAT("Ventas"."Unit price", 1, 1, 0) AS "Unit price", CONVERT_TO_INTEGER("Ventas"."Quantity", 1, 1, 0) AS Quantity, CONVERT_TO_FLOAT("Ventas"."Total", 1, 1, 0) AS Total
    FROM Ventas,Ventas AS Ventas
  ) nested_0
) nested_1;
```

Y mostraría este resultado:

Invoice ID	Invoice ID_1_2	Invoice ID_1_1	Product line	Unit price	Quantity	Total
750-67-8428	750_67_8428	750	Health And Beauty	76.69	7	546.9715
226-31-3081	226_31_3081	226	Electronic Accessories	25.28	5	86.22
631-41-3108	631_41_3108	631	Home And Lifestyle	46.33	7	348.5255
123-19-1176	123_19_1176	123	Health And Beauty	58.22	8	485.848
373-73-7910	373_73_7910	373	Sports And Travel	86.31	7	634.3785
699-14-3026	699_14_3026	699	Electronic Accessories	85.39	7	627.6165
355-53-5943	355_53_5943	355	Electronic Accessories	88.84	6	433.092
315-22-5665	315_22_5665	315	Home And Lifestyle	72.56	18	772.36
665-32-9167	665_32_9167	665	Health And Beauty	76.36	7	76.146
692-92-5582	692_92_5582	692	Food And Beverages	54.84	3	172.746
351-62-0822	351_62_0822	351	Fashion Accessories	54.48	4	68.816
529-56-3974	529_56_3974	529	Electronic Accessories	25.51	4	107.142
365-64-0515	365_64_0515	365	Electronic Accessories	46.95	5	246.8075
252-56-2699	252_56_2699	252	Food And Beverages	43.19	18	451.495
809-34-3928	809_34_3928	809	Health And Beauty	71.38	16	742.449
299-46-1885	299_46_1885	299	Sports And Travel	93.72	6	598.436
656-95-9349	656_95_9349	656	Health And Beauty	68.93	7	586.6355
765-26-6951	765_26_6951	765	Sports And Travel	72.61	6	457.443
329-62-1586	329_62_1586	329	Food And Beverages	54.67	3	172.2185
319-58-1348	319_58_1348	319	Home And Lifestyle	48.3	2	84.63

Vamos a guardarlo como un dataset en el espacio de ventas:

Save View As

Name

VentasEditadas

Location

> @Pedro

▼ Ventas

Datasets		Ventas
Pedro		Name
▼ Spaces (1)	⊕	Ventas
Ventas	2	VentasEditadas
Sources	⊕	
▼ Object Storage (1)		
Samples	0	

Copy Path