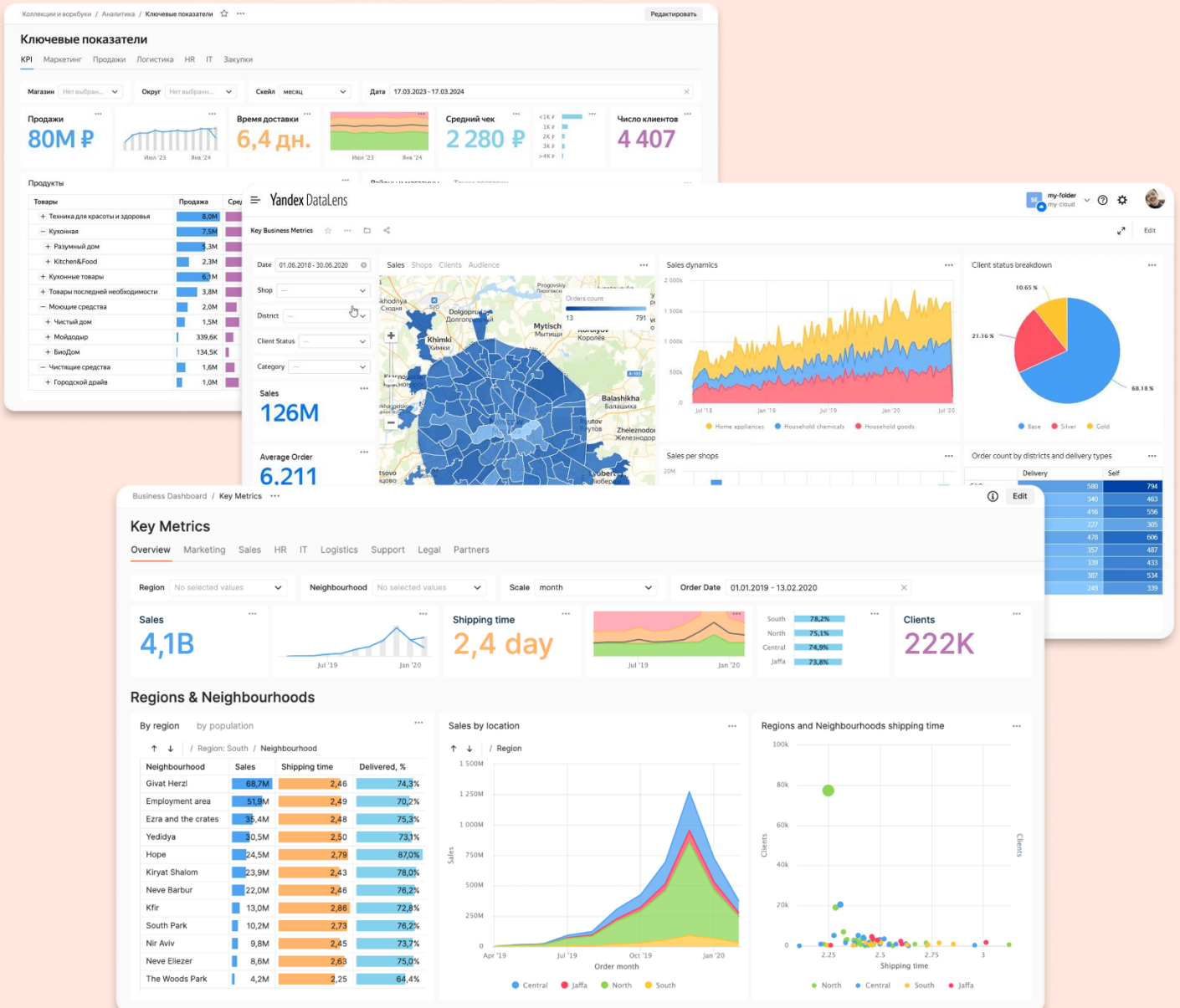




DataLens

Analytics at any scale. Now open source.



CONTENIDO

1.	INTRODUCCIÓN	3
2.	CONCEPTOS BÁSICOS.....	4
3.	CASO PRÁCTICO	9
4.	COMPARACIÓN CON POWERBI	23
5.	CONCLUSIONES	25

1. INTRODUCCIÓN

DataLens es un sistema de Business Intelligence y de visualización de datos de código abierto del equipo de Yandex Cloud. Permite conectarse a varias fuentes de datos para visualizar los datos y crear dashboards mediante una interfaz de usuario accesible.

Como aplicación de código abierto, puede instalarse rápidamente mediante contenedores de Docker o bien se puede utilizar la versión Cloud por parte de Yandex Cloud. <https://cloud.yandex.com/en/services/datalens>

En este documento nos centramos en la aplicación de código abierto. La podemos encontrar en <https://github.com/datalens-tech/datalens>

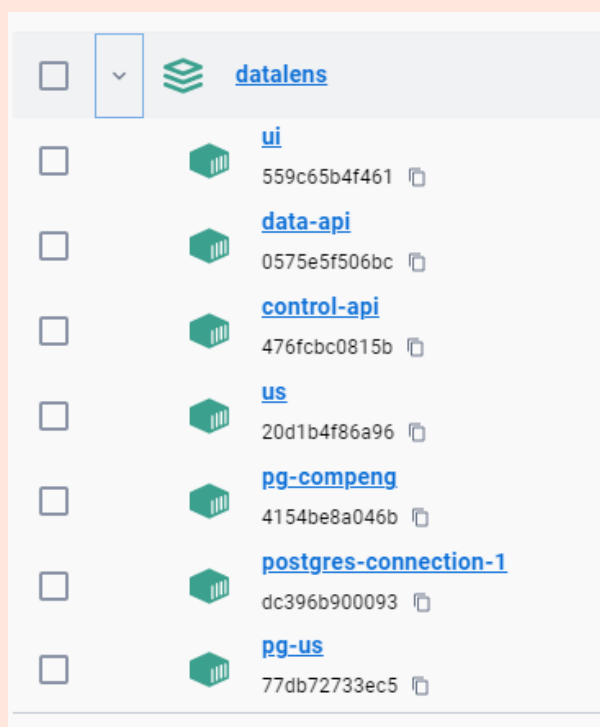
Para utilizarla, es necesario tener Docker instalado en nuestras máquinas. Con el siguiente comando se lanzan los contenedores de DataLens.

```
git clone https://github.com/datalens-tech/datalens && cd datalens
```

```
HC=1 docker compose up
```

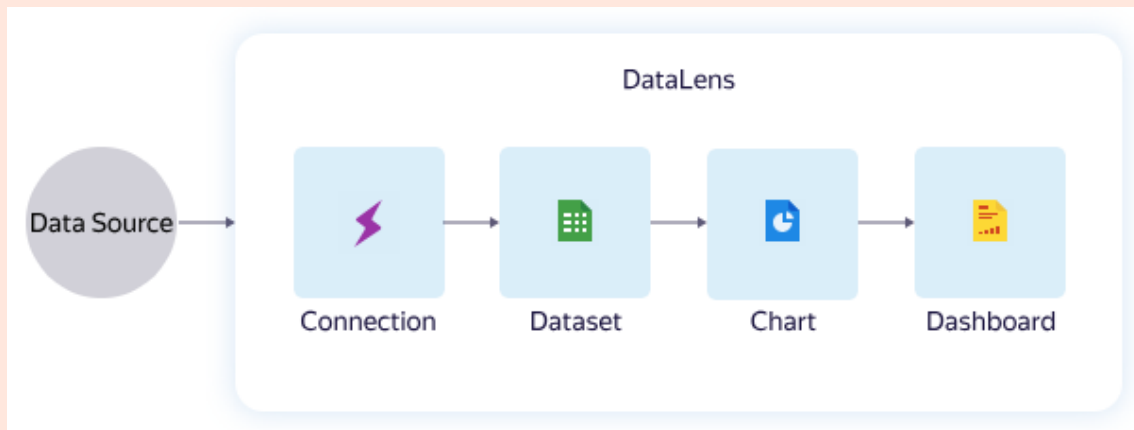
DataLens está formado por 3 partes principales:

- UI: aplicación que provee de la interfaz de usuario.
- Backend: conjunto de aplicaciones y librerías de Python para conectar a las fuentes de datos, generar consultas y procesar los datos.
- UnitedStorage (US): servicio que utiliza PostgreSQL para almacenar los metadatos y configuración de los objetos de DataLens.



2. CONCEPTOS BÁSICOS

DataLens utiliza varias entidades para el ciclo de datos completo:



A partir de la fuente de datos (Data Source), se definen las siguientes entidades:

- **Connection (conexión):** definición de la conexión a la fuente de datos. Por defecto, los conectores disponibles son PostgreSQL, ClickHouse y CHYT.
- **Dataset (conjunto de datos):** descripción del conjunto de datos de la fuente de datos.
- **Chart (gráfico):** datos del Dataset visualizados como tablas, diagramas o gráficos.
- **Dashboard (cuadro de mando):** conjunto de Charts, selectores para filtrar datos y bloques de texto.

Estas entidades se definen en un Workbook, que se organizan en Collections, contenedores que agrupan otras Workbooks y otras Collections.

Como ejemplo de estos conceptos, utilizamos la demo que viene por defecto en los contenedores.

Conexiones

El formulario muestra la configuración para conectar con ClickHouse. Incluye campos para Hostname, HTTP interface port, Username y Password. También hay opciones para Cache TTL in seconds (Default o Customized) y Raw SQL level (Forbid, Allow subselects in datasets, o Allow subselects in datasets and queries from charts). En la parte inferior, hay un botón 'Check connection' y un botón 'Save changes'.

ClickHouse

Hostname: rc1a-dcvk9cgrhmu889sj.mdb.yandexcloud.net,rc1b-lk2m3n43pststo6 X

HTTP interface port: 8443 X

Username: opensource-demo X

Password: X

Cache TTL in seconds: Default Customized

Raw SQL level ?
☐ Forbid
☐ Allow subselects in datasets
This option allows you to describe dataset sources using SQL queries
☒ Allow subselects in datasets and queries from charts
This option allows you to describe dataset sources using SQL queries and create QL charts

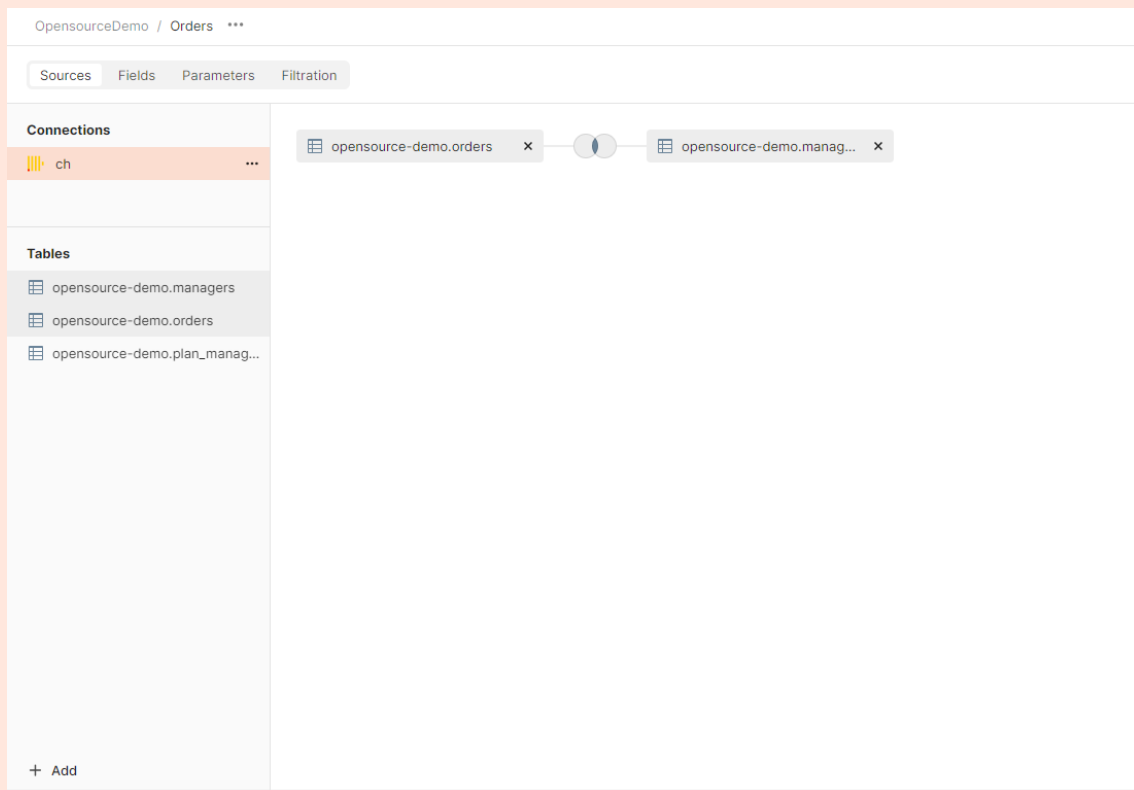
Advanced connection settings ▼

Check connection Save changes

En este caso, es una conexión a ClickHouse. Los parámetros de configuración son bastante estándar en la definición de conexiones a bases de datos (nombre del host, puerto, usuario, contraseña...).

Dataset

Los datos en el dataset se representan como campos. DataLens crea un dataset basado en una fuente de datos que tiene una conexión. Se utiliza en modo de acceso directo (Direct Query en PowerBI), se ejecutan las consultas en la fuente.



En el apartado Sources del dataset Orders, bajo Connections encontramos la conexión anterior “ch”, y bajo Tables encontramos tres tablas de la conexión. A la derecha se comprueban las relaciones entre tablas (entre “orders” y “managers”).

OpensourceDemo / Orders ***

Sources Fields Parameters Filtration

Update fields Preview + Add field Field name

#	Name	Field source	Type	Aggregation	Description
1	Order date (scale)	f(x)	Date	No	
2	Shipping time	f(x)	Fractional number	Average	
3	Order month	f(x)	Date	No	
4	Order year	f(x)	Integer	No	
5	Neighbourhood Polygon	orders.order_buyer_region_province_polygon	Geopolygon	No	
6	Delivered, %	f(x)	Fractional number	Auto	
7	Order ID	orders.order_id	Integer	No	
8	Orders	orders.order_id	Integer	Count (Distinct)	
9	Client ID	orders.user_id	Integer	No	
10	Clients	orders.user_id	Integer	Count (Distinct)	
11	Order date and time	orders.order_creation_date	Date and time	No	
12	Order date	orders.order_creation_date	Date	No	
13	Delivery date and time	orders.order_delivery_datetime	Date and time	No	
14	Delivery date	orders.order_delivery_datetime	Date	No	
15	Quantity	orders.quantity	Integer	Sum	
16	Delivery service	orders.order_delivery_attribute_delivery_service...	String	No	
17	Payment type	orders.order_delivery_attribute_delivery_service...	String	No	

En el apartado Fields podemos ver los campos definidos para este dataset. De izquierda a derecha vemos los apartados: # (número de campo), Name (nombre de campo), Field source (origen del campo, puede ser un campo de alguna de las tablas anteriores o un campo calculado que se muestra como f(x)), Visibilidad, Type (tipo del dato, puede ser fecha, fecha y hora, número entero o fracción, polígono, cadena...), Aggregation (función de agregación del campo) y Description (descripción del campo).

OpensourceDemo / Orders ***

Sources Fields Parameters Filtration

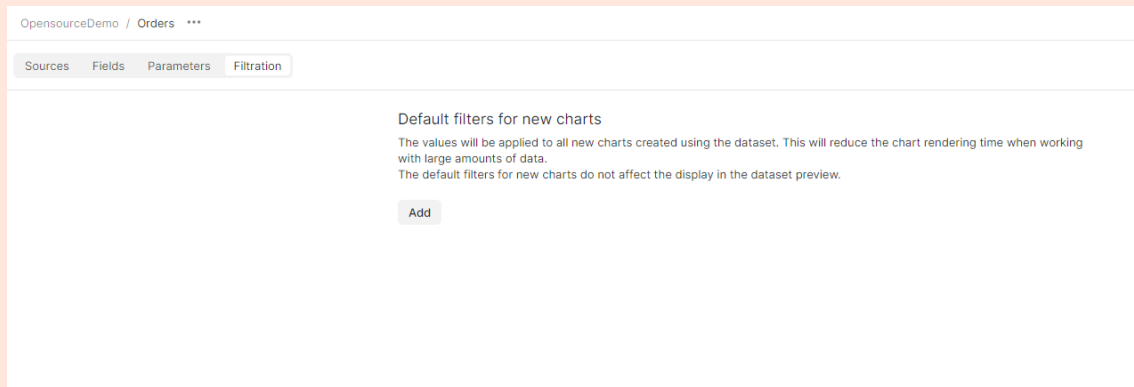
Parameters

A parameter is a dataset or chart variable that can replace constant values in calculated fields. The parameter can be a number, string, date, or boolean. Dataset parameters are available in all charts based on this dataset. A default parameter value can be overridden at the chart level.

Name	Type	Default Value
metric	String	sales
ILS-USD	Fractional number	0.25
ILS-EUR	Fractional number	0.27
mavg-value	Integer	7
dimension	String	Region
currency	String	ILS
ago-period	Integer	1
scale	String	month

Add

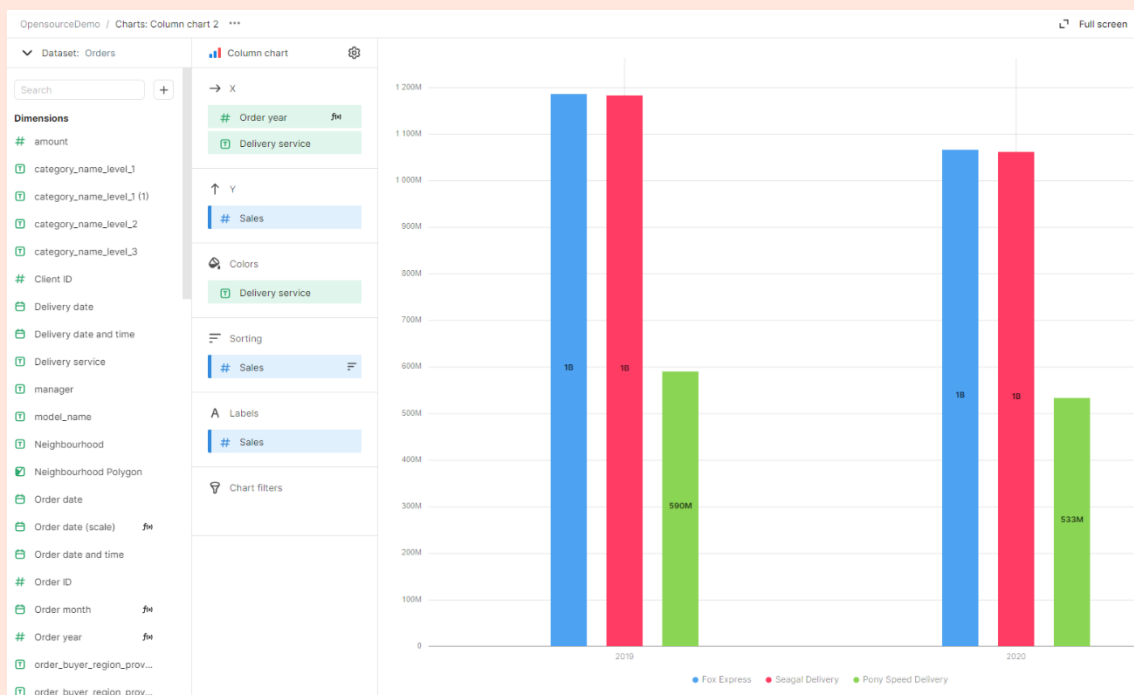
En el apartado de Parameters se encuentran algunos parámetros definidos a nivel de Dataset.



En el apartado de Filtration se pueden definir filtros a nivel de Dataset para evitar problemas de rendimiento o limitaciones técnicas si la cantidad de datos es muy alta.

Charts

En la demo encontramos ejemplos de todos los tipos de Charts que se pueden definir. Por ejemplo, un gráfico de columnas:



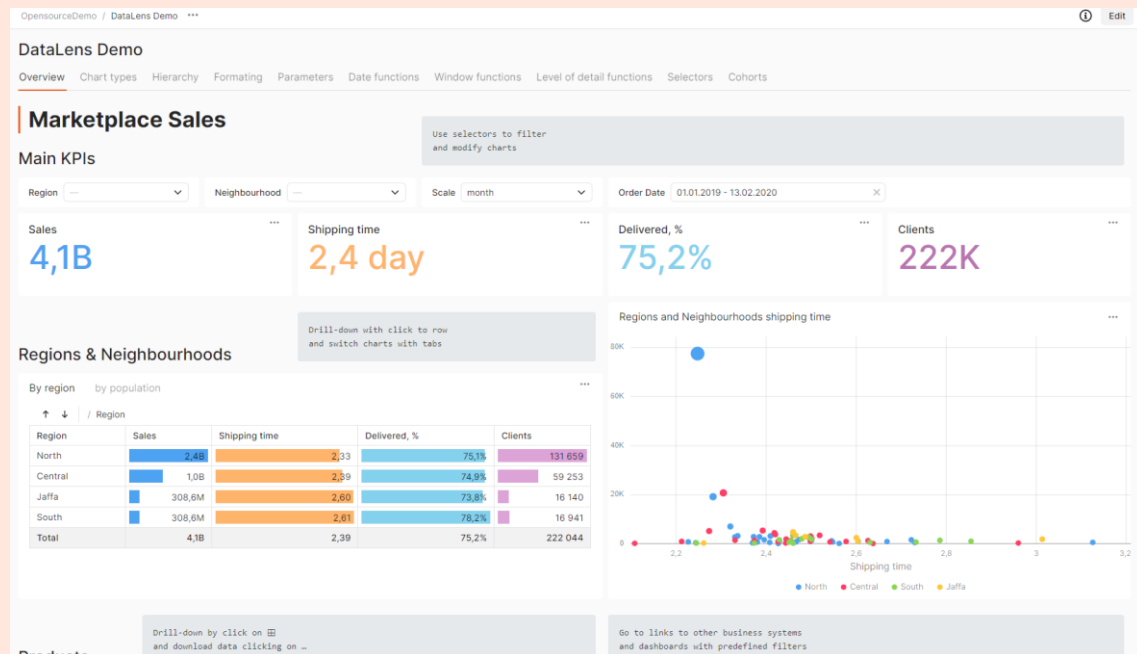
A la izquierda encontramos el Dataset que queremos utilizar para el Chart. Bajo el apartado “Column chart” vemos la definición del gráfico:

- **X:** definición del eje X. En este caso, Order year y delivery service.
- **Y:** Número de ventas.
- **Colors:** colores para distinguir los datos, en este caso se utiliza el campo Delivery service.

- **Sorting:** campo por el que se ordenan los datos. En este caso es directamente las ventas de manera descendiente.
- **Labels:** Etiquetas de los datos, en este caso son las ventas.

Dashboards

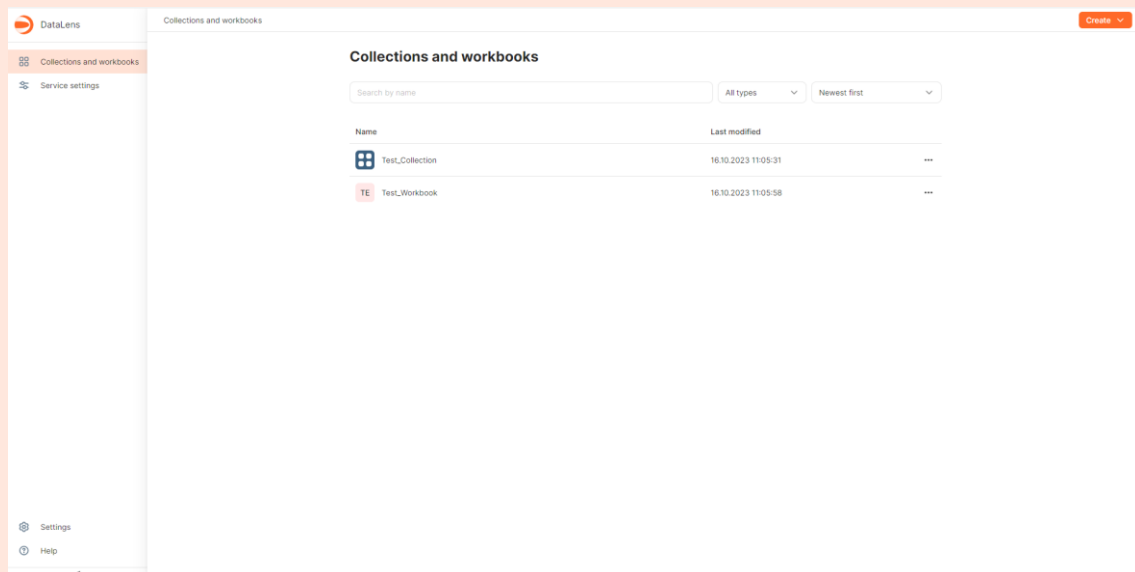
Finalmente



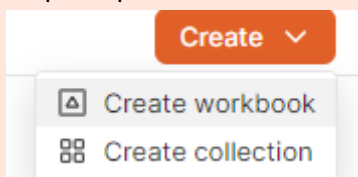
Finalmente, tenemos el Dashboard de la demo, donde se pueden ver los Charts previamente definidos, selectores para filtrar los datos (Region, Neighbourhood, Scale...) y bloques de texto.

3. CASO PRÁCTICO

Veremos un pequeño caso práctico en el cual diseñaremos un pequeño dashboard con los datos de un modelo en estrella alojados en una base de datos de PostgreSQL.



En la página principal de DataLens hacemos click en Create y seleccionamos Create

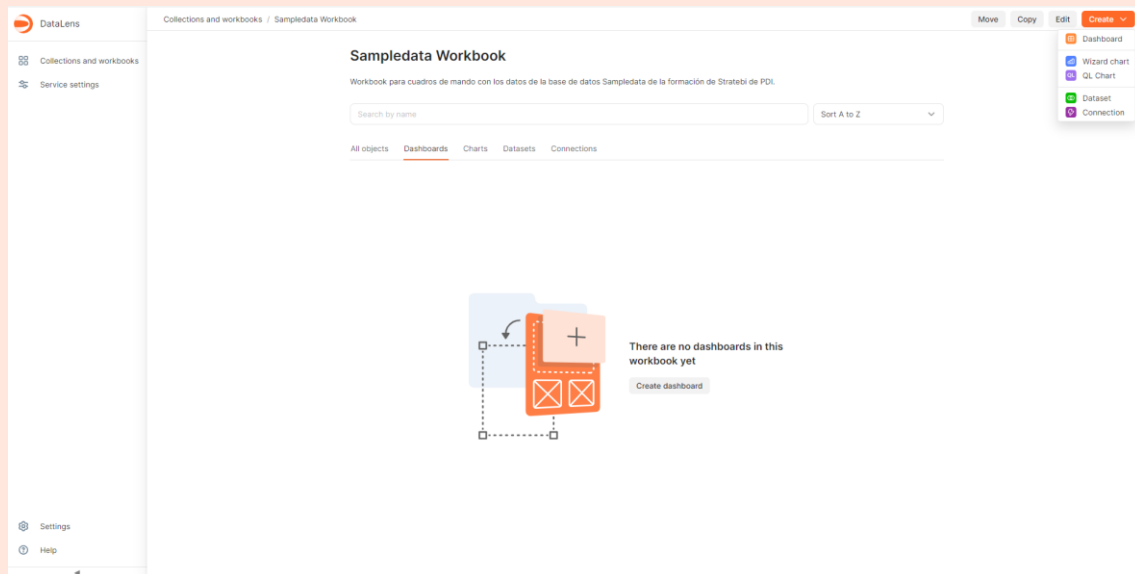


workbook.

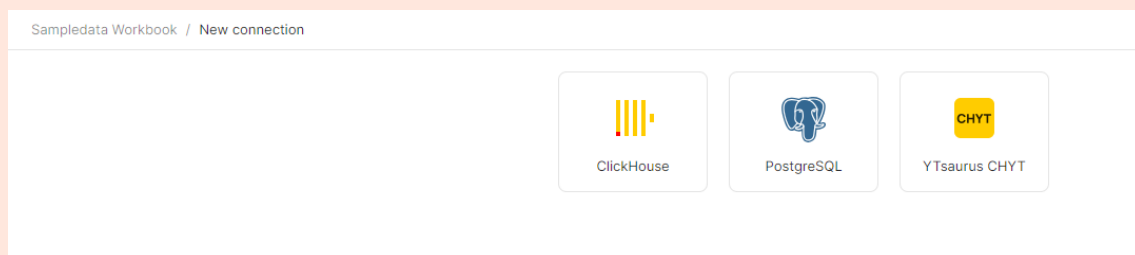
Añadimos un nombre y una descripción (opcional) y hacemos click en Create.

A screenshot of the 'Create workbook' dialog box. It has a title bar with 'Create workbook' and a close button. The form contains two fields: 'Name' with the value 'Sampledata Workbook' and 'Description' with the value 'Workbook para cuadros de mando con los datos de la base de datos Sampledata de la formación de Stratebi de PDI.'. At the bottom, there are two buttons: 'Cancel' and 'Create'.


Dentro del workbook, volvemos a hacer click en Create. Empezamos creando la conexión a la base de datos, haciendo click en Connection.



Aquí aparecen los conectores existentes. Como comentábamos, la versión de código abierto tiene por defecto los conectores ClickHouse, PostgreSQL y CHYT, aunque es posible agregar más mediante código.



En nuestro caso, es una BD de PostgreSQL. Definimos todos los parámetros. En concreto, seleccionamos la opción “Allow subselects in datasets and queries from charts”. Hacemos click en Check connection para verificar que la definición de la conexión es correcta y que DataLens tiene acceso a la base de datos.

←  PostgreSQL

Hostname

host.docker.internal

×

Port

5432

×

Path to database

postgres

×

Username

postgres

×

Password


••••••

×

Cache TTL in seconds

Default

Customized

Raw SQL level 

☐ Forbid

☐ Allow subselects in datasets
This option allows you to describe dataset sources using SQL queries

☒ Allow subselects in datasets and queries from charts
This option allows you to describe dataset sources using SQL queries and create QL charts

Advanced connection settings

▼

Check connection

✓

Create connection

Al hacer click en Create connection, le damos nombre a la conexión. Dentro del workbook, verificamos que la conexión está creada en el apartado Connections.

Sampledata Workbook

Workbook para cuadros de mando con los datos de la base de datos Sampledata de la formación de Stratebi de PDI.

Search by name

Sort A to Z

▼


All objects

Dashboards

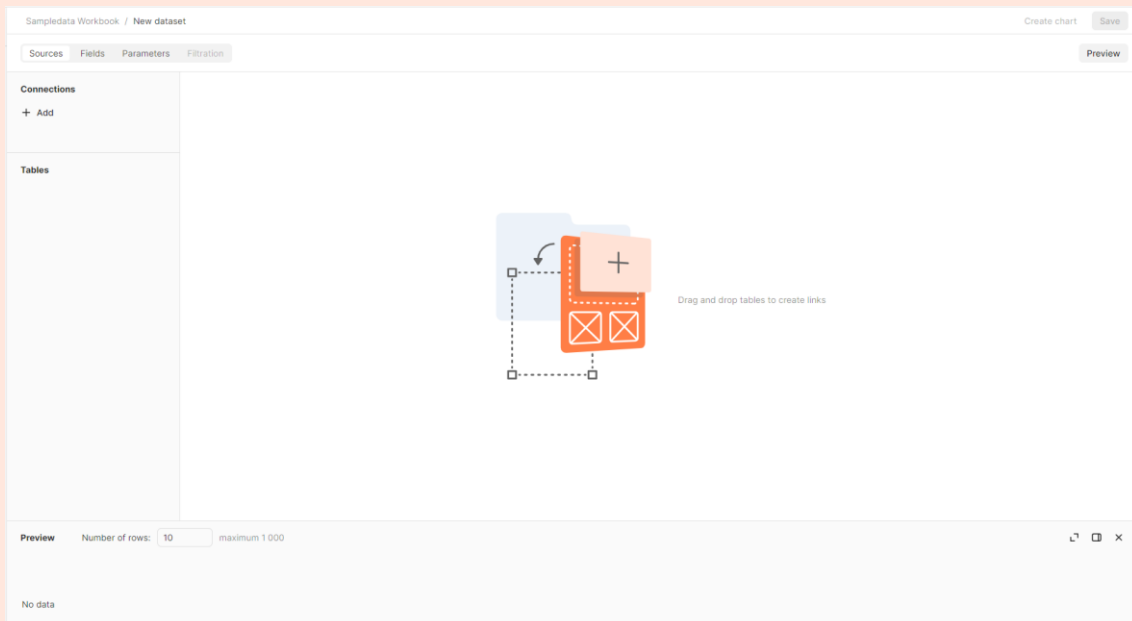
Charts

Datasets

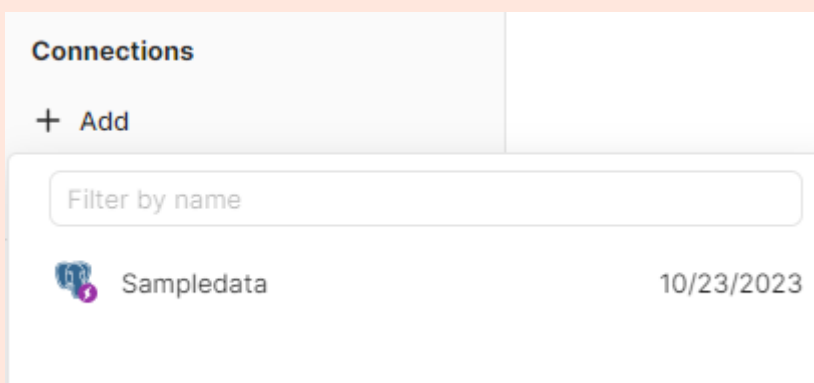
Connections

Name	Last modified	
 Sampledata	23.10.2023 10:36:29	...

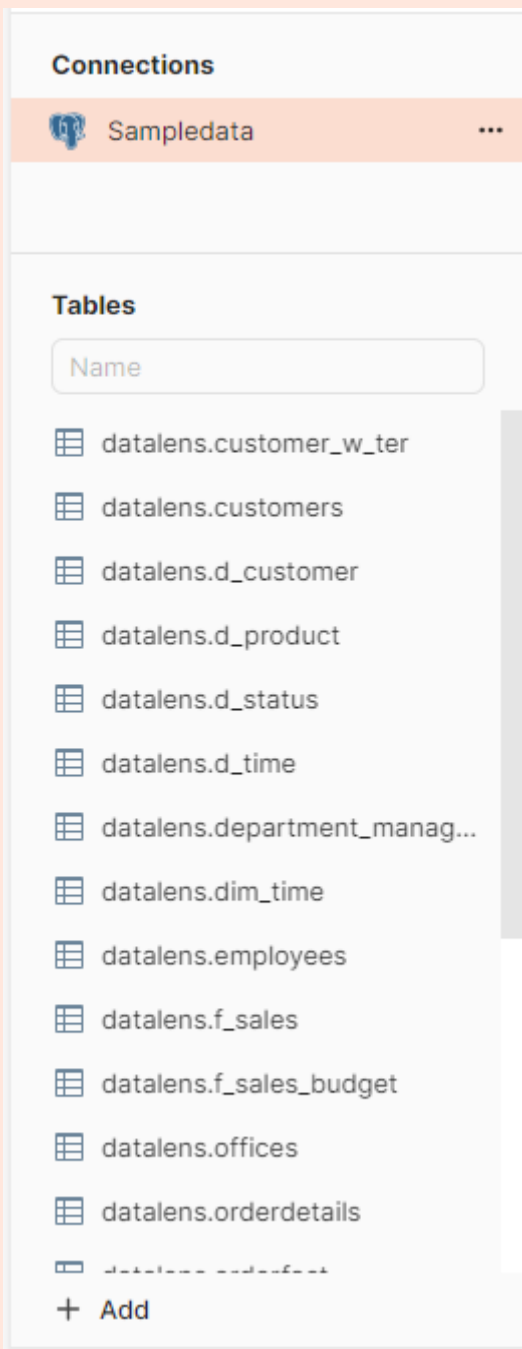
Ahora, procedemos a crear un Dataset. Volvemos a hacer click en Create y seleccionamos Dataset.



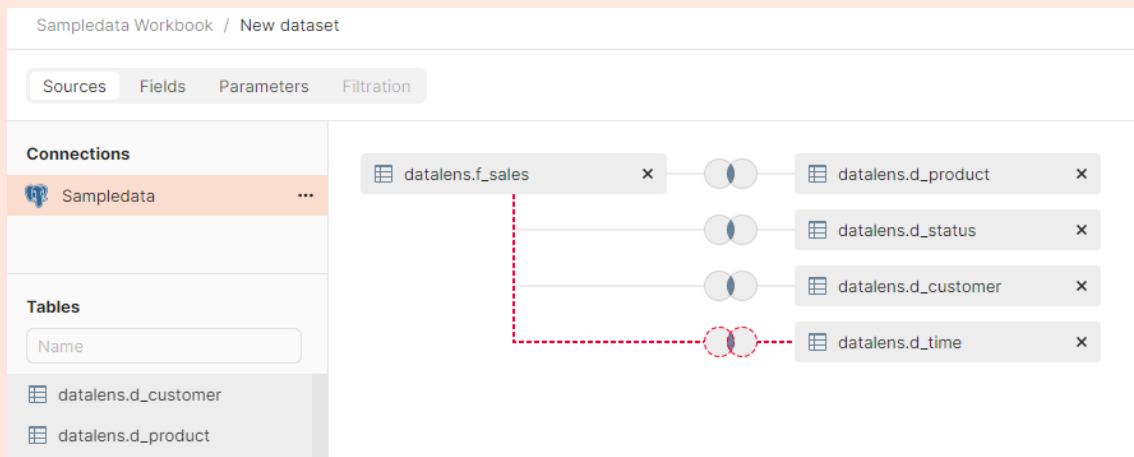
Añadimos la conexión que hemos definido haciendo click en “+ Add” debajo de Connections.



Al añadir la conexión, automáticamente aparecen las tablas de la base de datos.



Arrastramos las tablas al panel de la derecha y definimos las relaciones en caso de que no se autodetecten. En este caso, DataLens no detecta una relación directa entre f_sales y d_time y aparece con una línea roja discontinua para mostrar que existe un error de asignación.



Hacemos click en el diagrama de Venn de la relación. Aquí podemos definir el tipo de join (Inner, Left, Right, Full) y las columnas por las que hay que relacionar con “Add Link”.

Link X

datalens.f_sales --- Venn --- datalens.d_time

Add link

Cancel Apply

Link X

datalens.f_sales --- Venn --- datalens.d_time

ID_ORDER_DATE ▾ = # ID_TIME ▾

Add link

Cancel Apply

Una vez todas las relaciones están correctamente configuradas, en el apartado inferior aparece una vista previa de los datos.

Sampledata Workbook / New dataset Create chart Save

Sources Fields Parameters Filtration Preview

Connections

Sampledata ...

Tables

Name

- dataens.d_time
- dataens.f_sales
- dataens.customer_w_ter
- dataens.customers
- dataens.department_manag...
- dataens.dim_time
- dataens.employees
- dataens.f_sales_budget
- dataens.offices
- dataens.orderdetails
- dataens.orderfact
- dataens...

+ Add

Preview Number of rows: 10 maximum 1000

#	ID_CUSTOMER	ID_PRODUCT	ID_STATUS	ID_ORDER_DATE	ID_REQUIRED_DATE	ID_SHIPPED_DATE	QUANTITYORDERED	AMOUNT	ID_CUSTOMER (1)	DES_CUSTOMER	CREDITLIMIT	ID_PRODUCT (1)	VERSION	DATE_FROM
1	103	3	6	20040927	20041005	20041001	39	96.0	103	Atelier graphique	21000	3	1	1900-01-01T00:00:00
2	103	21	6	20030520	20030529	20030522	26	118.0	103	Atelier graphique	21000	21	1	1900-01-01T00:00:00
3	103	32	6	20040927	20041005	20041001	32	48.0	103	Atelier graphique	21000	32	1	1900-01-01T00:00:00
4	103	34	6	20030520	20030529	20030522	46	112.0	103	Atelier graphique	21000	34	1	1900-01-01T00:00:00
5	103	46	6	20030520	20030529	20030522	34	157.0	103	Atelier graphique	21000	46	1	1900-01-01T00:00:00

En caso de querer añadir alguna subconsulta, hacemos click en “+ Add” debajo de Tables. Esto puede ser útil por ejemplo para pre-filtrar algunos resultados y mejorar el rendimiento de la carga de los gráficos si la tabla de hechos es muy grande.

Source ×

Source name

SQL

Subquery ?

```
1 select 1 as a
```

Cancel Create

En el apartado de Fields, vamos a definir los campos correctamente. En concreto, vamos a eliminar los ID repetidos y a definir las agregaciones de los campos de métricas.

Sampledata Workbook / New dataset Create chart Save

Sources Fields Parameters Filtration Update fields Preview + Add field

#	Name ↓	Field source ↓	Type ↓	Aggregation ↓	Description ↓
1	ID_CUSTOMER	f_sales.ID_CUSTOMER	# Integer	No	
2	ID_PRODUCT	f_sales.ID_PRODUCT	# Integer	No	
3	ID_STATUS	f_sales.ID_STATUS	# Integer	No	
4	ID_ORDER_DATE	f_sales.ID_ORDER_DATE	# Integer	No	
5	ID_REQUIRED_DATE	f_sales.ID_REQUIRED_DATE	# Integer	No	
6	ID_SHIPPED_DATE	f_sales.ID_SHIPPED_DATE	# Integer	No	
7	QUANTITYORDERED	f_sales.QUANTITYORDERED	# Integer	No	
8	AMOUNT	f_sales.AMOUNT	# Fractional number	No	
9	ID_CUSTOMER (1)	d_customer.ID_CUSTOMER	# Integer	No	
10	DES_CUSTOMER	d_customer.DES_CUSTOMER	String	No	
11	CREDITLIMIT	d_customer.CREDITLIMIT	# Integer	No	
12	ID_PRODUCT (1)	d_product.ID_PRODUCT	# Integer	No	
13	VERSION	d_product.VERSION	# Integer	No	
14	DATE_FROM	d_product.DATE_FROM	Date and time	No	
15	DATE_TO	d_product.DATE_TO	Date and time	No	
16	ID_PRODUCT_NK	d_product.ID_PRODUCT_NK	String	No	

Al colocar el ratón encima de algún campo, aparecen 3 puntos a la derecha. Al hacer click en ellos, podemos eliminar los campos que queramos.

9	ID_CUSTOMER (1)	d_customer.ID_CUSTOMER	# Integer	No	...
10	DES_CUSTOMER	d_customer.DES_CUSTOMER	String	No	
11	CREDITLIMIT	d_customer.CREDITLIMIT	# Integer	No	
12	ID_PRODUCT (1)	d_product.ID_PRODUCT	# Integer	No	

Duplicate
Edit
Copy ID
Delete

También podemos cambiar la agregación en el apartado “Aggregation”.

#	Name ↓	Field source ↓	Type ↓	Aggregation ↓	Description ↓
1	ID_CUSTOMER	f_sales.ID_CUSTOMER	# Integer	No	
2	ID_PRODUCT	f_sales.ID_PRODUCT	# Integer	No	
3	ID_STATUS	f_sales.ID_STATUS	# Integer	No	
4	ID_ORDER_DATE	f_sales.ID_ORDER_DATE	# Integer	No	
5	ID_REQUIRED_DATE	f_sales.ID_REQUIRED_DATE	# Integer	No	
6	ID_SHIPPED_DATE	f_sales.ID_SHIPPED_DATE	# Integer	No	
7	QUANTITYORDERED	f_sales.QUANTITYORDERED	# Integer	No	
8	AMOUNT	f_sales.AMOUNT	# Fractional number	No	
9	DES_CUSTOMER	d_customer.DES_CUSTOMER	String	No	
10	CREDITLIMIT	d_customer.CREDITLIMIT	# Integer	No	
11	VERSION	d_product.VERSION	# Integer	No	
12	DATE_FROM	d_product.DATE_FROM	Date and time	No	
13	DATE_TO	d_product.DATE_TO	Date and time	No	
14	ID_PRODUCT_NK	d_product.ID_PRODUCT_NK	String	No	

No
Average
Count
Count (Distinct)
Maximum
Minimum
Sum

Por último, para crear un campo calculado, hacemos click en “+ Add Field”. Aquí podemos crear métricas adicionales. Las funciones disponibles están descritas en la documentación.

Field settings

NUM_PRODUCTS

Formula

Field from source

Don't display in wizard

Field

1 COUNTD([ID_PRODUCT])

ID_CUSTOMER

ID_PRODUCT

ID_STATUS

ID_ORDER_DATE

ID_REQUIRED_DATE

ID_SHIPPED_DATE

QUANTITYORDERED

AMOUNT

DES_CUSTOMER

CREDITLIMIT

VERSION

DATE_FROM

DATE_TO

ID_PRODUCT_NK

DES_PRODUCT

DES_PRODUCT_LINE

ID_PRODUCT_LINE

DES_STATUS

ID_STATUS_NK

Cancel

Create

Cuando tenemos el resultado final, hacemos click en Save y le damos nombre.

Sampledata Workbook / F_Sales

Create chart

Save

Sources

Fields

Parameters

Filtration

Update fields

Preview

+ Add field

Field name

#	Name	Field source	Type	Aggregation	Description
1	NUM_PRODUCTS	f_sales.ID_PRODUCT	# Integer	Auto	
2	ID_CUSTOMER	f_sales.ID_CUSTOMER	# Integer	No	
3	ID_PRODUCT	f_sales.ID_PRODUCT	# Integer	No	
4	ID_STATUS	f_sales.ID_STATUS	# Integer	No	
5	ID_ORDER_DATE	f_sales.ID_ORDER_DATE	# Integer	No	
6	ID_REQUIRED_DATE	f_sales.ID_REQUIRED_DATE	# Integer	No	
7	ID_SHIPPED_DATE	f_sales.ID_SHIPPED_DATE	# Integer	No	
8	QUANTITYORDERED	f_sales.QUANTITYORDERED	# Integer	Sum	
9	AMOUNT	f_sales.AMOUNT	# Fractional number	Sum	
10	DES_CUSTOMER	d_customer.DES_CUSTOMER	String	No	
11	CREDITLIMIT	d_customer.CREDITLIMIT	# Integer	No	
12	VERSION	d_product.VERSION	# Integer	No	
13	DATE_FROM	d_product.DATE_FROM	Date and time	No	
14	DATE_TO	d_product.DATE_TO	Date and time	No	
15	ID_PRODUCT_NK	d_product.ID_PRODUCT_NK	String	No	
16	DES_PRODUCT	d_product.DES_PRODUCT	String	No	
17	DES_PRODUCT_LINE	d_product.DES_PRODUCT_LINE	String	No	

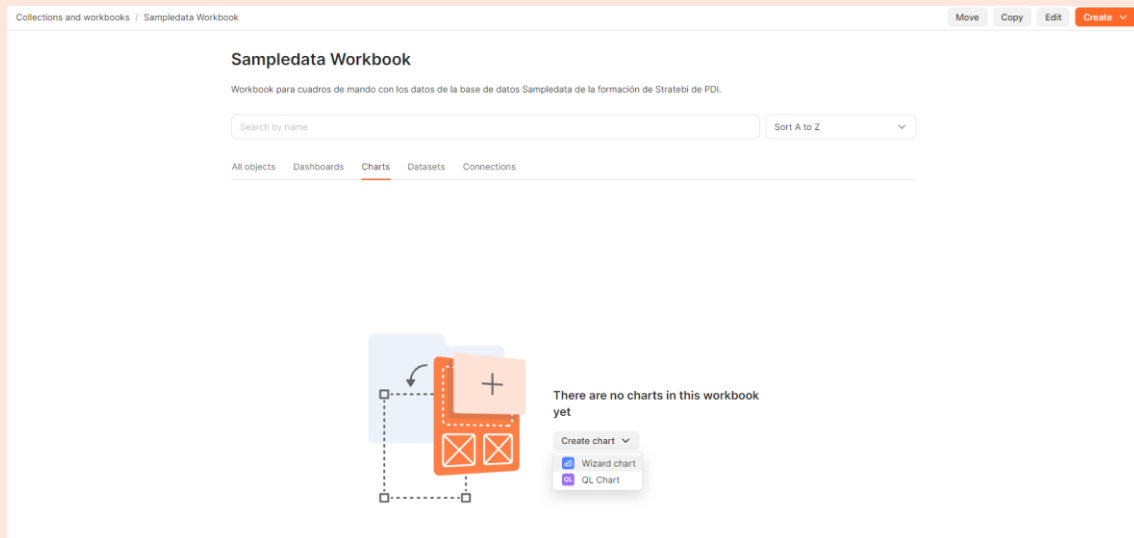
Preview

Number of rows: 10

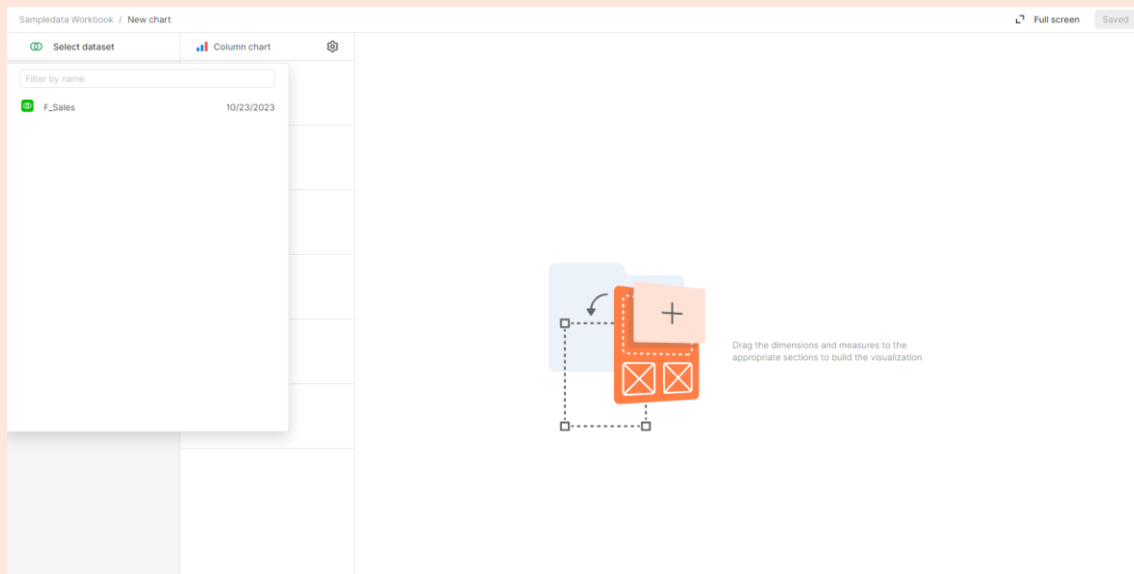
maximum 1 000

#	NUM_PRODUCTS	ID_CUSTOMER	ID_PRODUCT	ID_STATUS	ID_ORDER_DATE	ID_REQUIRED_DATE	ID_SHIPPED_DATE	QUANTITYORDERED	AMOUNT	DES_CUSTOMER	CREDITLIMIT	VERSION	DATE_FROM	DATE_TO
1	1	103	3	6	20040927	20041005	20041001	203	768.0	Atelier graphique	21000	1	1900-01-01T00:00:00	2199-12-31T23:59:59
2	1	103	21	6	20030520	20030529	20030522	86	944.0	Atelier graphique	21000	1	1900-01-01T00:00:00	2199-12-31T23:59:59
3	1	103	32	6	20040927	20041005	20041001	87	384.0	Atelier graphique	21000	1	1900-01-01T00:00:00	2199-12-31T23:59:59

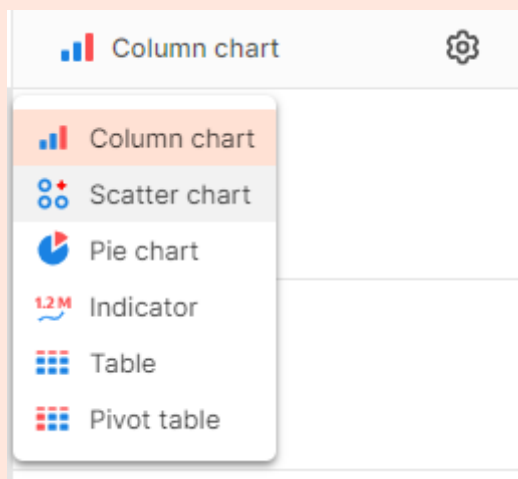
Una vez tenemos el dataset creado, vamos a crear dos gráficos simples. Para el primero vamos a darle a crear Wizard chart.



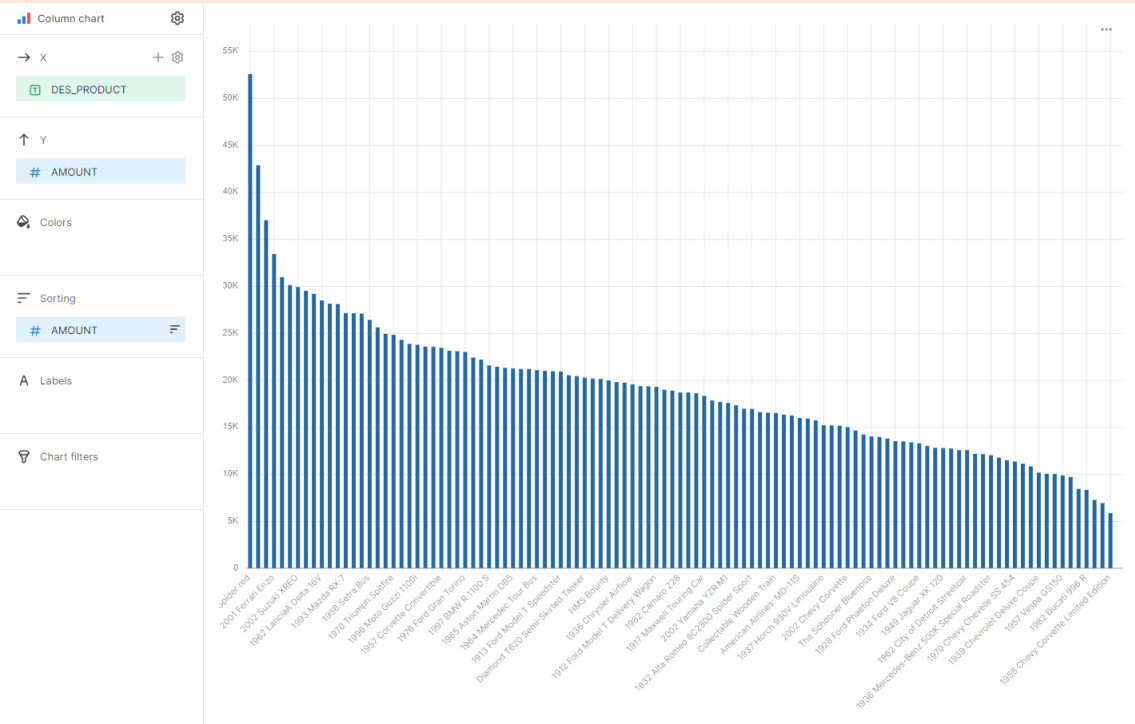
Esto nos lleva al Wizard de creación de gráficos donde podemos utilizar el dataset que hemos configurado.



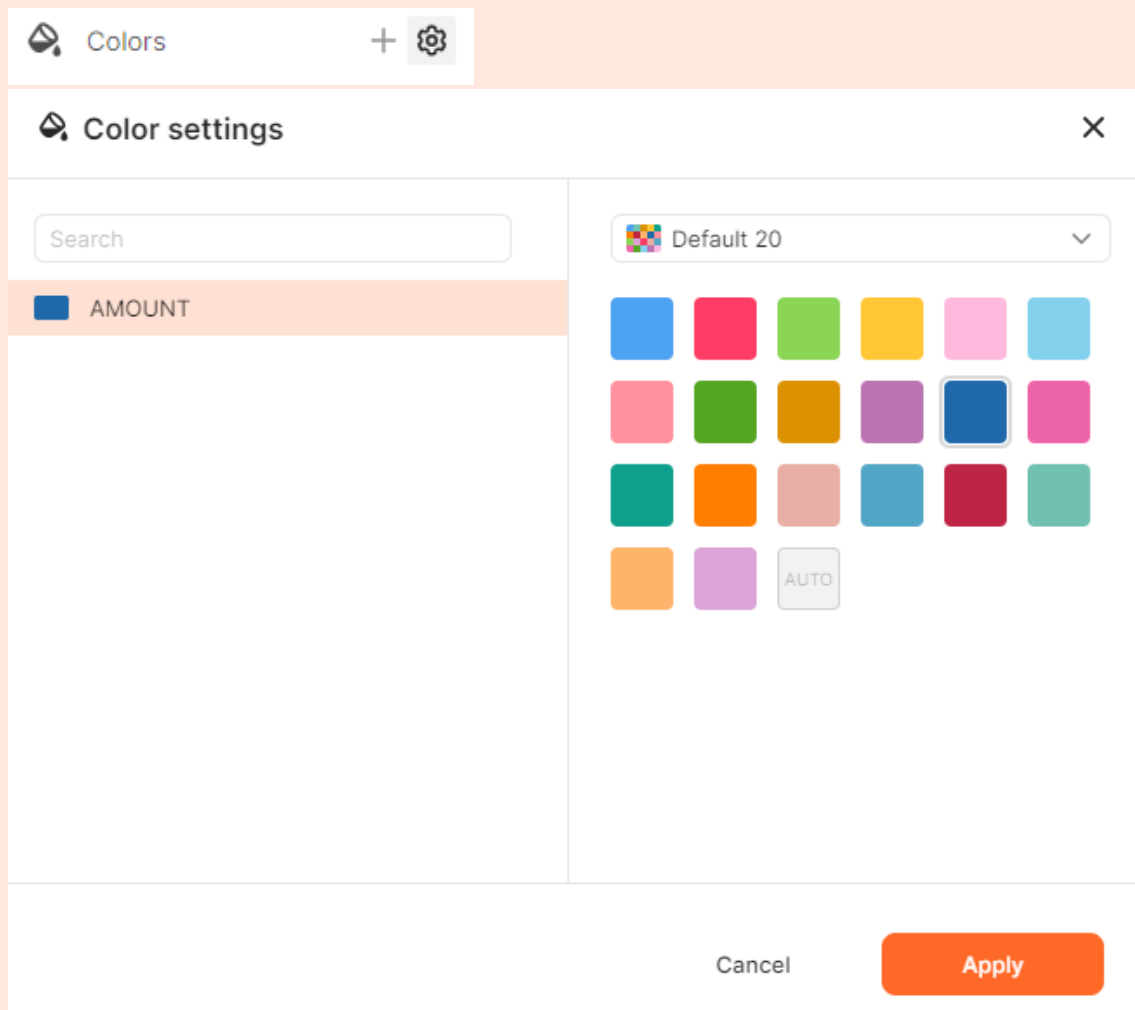
Aquí podemos elegir el tipo de gráfico:



Configuramos el gráfico simple arrastrando los campos a los apartados:



Para el color, hacemos click en la rueda y escogemos uno de los colores por defecto si no queremos utilizar un campo para dar colores:



Los ejes (X, Y) y las etiquetas (Labels) también tienen propiedades de configuración. Una vez configurado, le damos a Save para guardarlo.

La otra opción de creación de gráficos es “QL Chart”. Este gráfico hace consultas directas a la base de datos, por lo que no es necesario utilizar un Dataset. Se utilizan consultas Select para obtener los datos necesarios para el gráfico.

Hacemos click en Create -> QL Chart, elegimos la conexión y hacemos click en Create.

Sampledata Workbook / New chart

Create QL chart

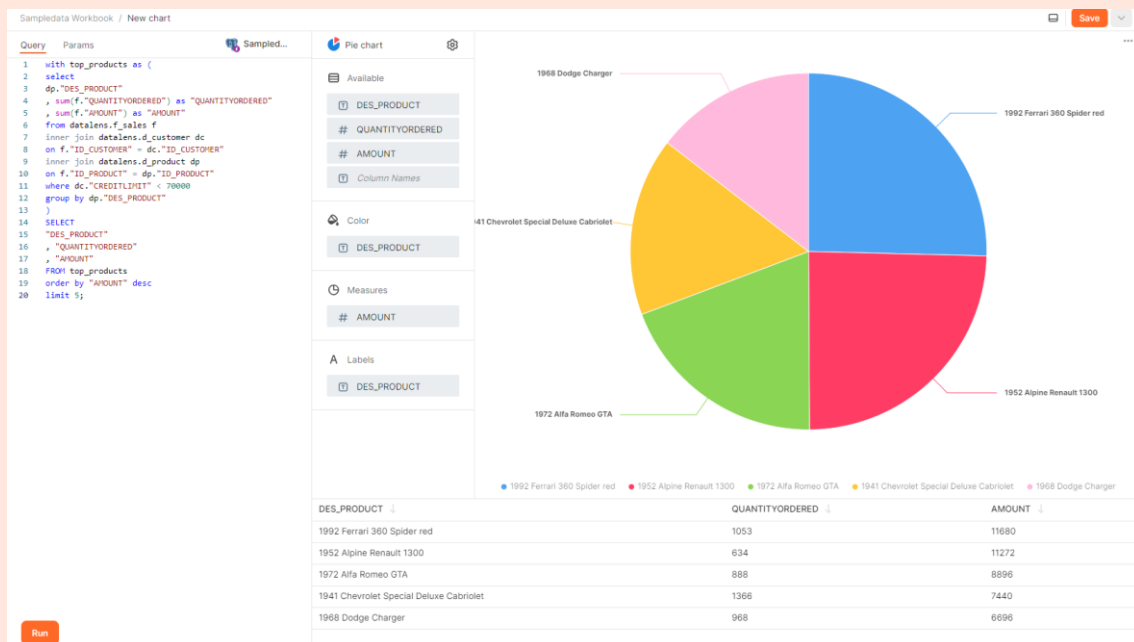
Chart type

SQL Prometheus Monitoring

Connection

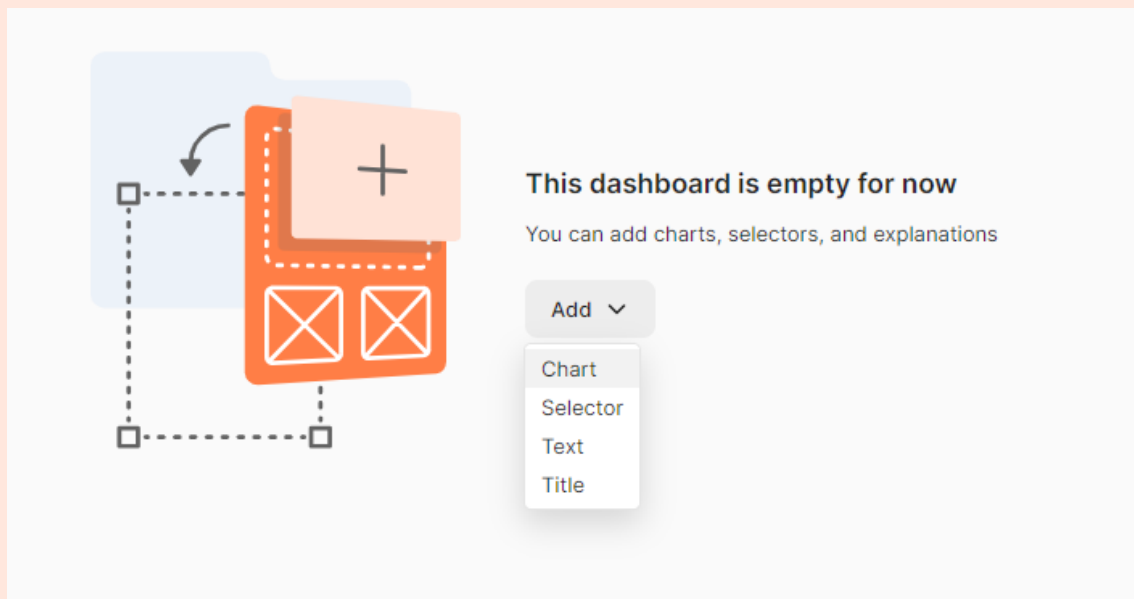
Create

Escribimos la query con los datos que necesitamos y le damos a Run. A la derecha nos aparecerán todos los campos de la query y podemos configurar el gráfico de manera parecida al gráfico anterior.



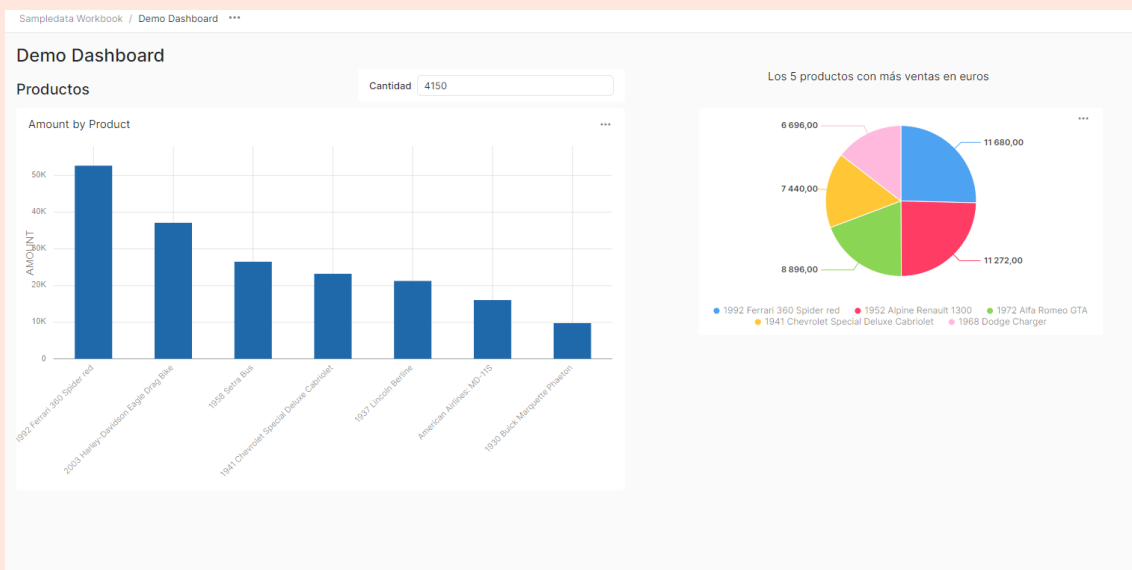
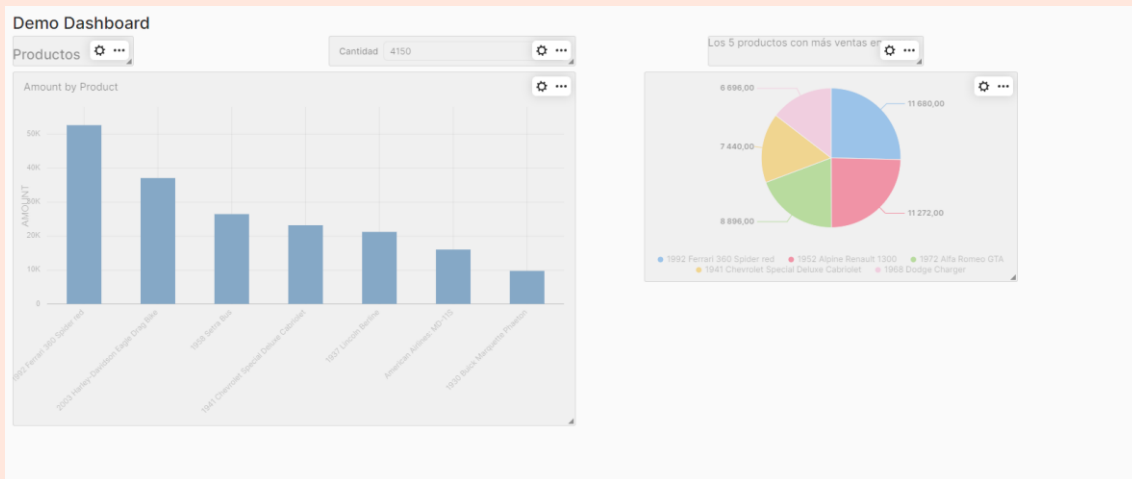
Las opciones de configuración del gráfico QL son mucho más limitadas. El beneficio de utilizar gráficos QL, entre otros, es aprovechar funciones específicas del dialecto de SQL utilizado en la base de datos.

Finalmente, pasamos a configurar un Dashboard. Hacemos click en Create Dashboard y le damos nombre. Empezamos por añadir los Charts que hemos creado:

This screenshot shows a 'Tab settings' dialog box. On the left, under the 'Chart' tab, there's a star icon and the title 'Amount by Product', followed by a '+ Add' button. The main area of the dialog is titled 'Tab settings' and contains the following fields: 'Title' (Amount by Product) with a 'Show' checkbox, 'Chart' (Amount by Product) with an 'Open' button, a 'Specify link' button, 'Description' (Enter description), and 'Auto height' (Enable) with a checkbox. At the bottom, there's a 'Parameters' section with a dropdown arrow, and 'Cancel' and 'Add' buttons.

Si queremos añadir el otro gráfico como otra pestaña, hacemos click en “+ Add” debajo del gráfico que acabamos de añadir. De lo contrario, habría que añadir un segundo gráfico.

Podemos añadir también selectores y texto. Al hacer click en Edit, se pueden mover los elementos para el diseño del cuadro de mando. En Tabs pueden añadirse más páginas.



4. COMPARACIÓN CON POWERBI

Característica	Power BI	DataLens
Visualización y reportes	X	X
Distintos tipos de visualizaciones	X	X
Cuadros de mando customizables	X	X
Exploración de datos interactiva	X	X
Preparación de datos y transformaciones	X	X
Modelado de datos	X	X
Cálculos avanzados (DAX)	X	

Colaborar y compartir	X	X
Integración con Microsoft Teams/SharePoint	X	
Gestión de accesos mediante roles	X	X
IA y Machine Learning	X	
Integración con machine learning	X	
Análisis mediante IA	X	
Seguridad y administración	X	X
Seguridad a nivel de fila	X	X
Características de gobierno de datos	X	
Integración con servicios Cloud	X	X
Ecosistema Microsoft	X	
Ecosistema Yandex Cloud		X
Soporte de archivos	X	X
Excel	X	X
PDF	X	
Soporte para modelos de datos grandes	X	
Parámetros condicionales (What-If)	X	
Monitorización y alerta de datos	X	
Análisis de datos geoespaciales	X	
Métricas de uso	X	

Consultas de lenguaje natural	X	
-------------------------------	---	--

5. CONCLUSIONES

La interfaz de usuario es bastante intuitiva y fácil de utilizar. Las opciones de configuración y diseño no son tan potentes como otras herramientas de reporting como PowerBI, pero para crear cuadros de mando funcionales de manera sencilla es una buena alternativa, especialmente si se hace uso del ecosistema de Yandex.

La versión de código abierto provee un buen punto de partida para ser utilizada como herramienta de reporting en una compañía si se posee un perfil desarrollador que pueda ampliarla. De lo contrario, existen bastantes limitaciones con las características por defecto, como los conectores a base de datos, que en la mayoría de los casos no serán suficientes. Además, la configuración de seguridad y usuarios también requiere de ese perfil.