Examen Transversal

**Docente:** Joel Enrique Pérez Contreras

**Integrantes:**

Camila Orozco

Gary Hernández

Darío Conuenao

**Fecha:** 15 Julio 2025

**Examen Trasversal**

**Caso Semestral “Taxis de New York” y “Subasta Streaming”.**

**BIG DATA\_001V**

Contenido

Contenido

[Introducción 3](#_Toc203357055)

[Parte 1: 4](#_Toc203357056)

[Arquitectura: 4](#_Toc203357057)

[1. Creación de Buckets: 5](#_Toc203357058)

[2. Carga datos: 6](#_Toc203357059)

[3. Creación de BigQuery 7](#_Toc203357060)

[4. Creación y carga de parquet 9](#_Toc203357061)

[5. Visualizacion 10](#_Toc203357062)

[6. Limpieza de datos 11](#_Toc203357063)

[7. Datos: 12](#_Toc203357064)

[8. Creacion de nuevas tablas. 14](#_Toc203357065)

[9. Comprobar nuevos datos 16](#_Toc203357066)

[10. Gráficos: 17](#_Toc203357067)

[Parte 2: 18](#_Toc203357068)

[Arquitectura: 18](#_Toc203357069)

[1. Creación del Webhook 19](#_Toc203357070)

[2. Configuración del tópico en Pub/Sub 20](#_Toc203357071)

[3. Desarrollo y despliegue en Cloud Run 21](#_Toc203357072)

[4. Registro del Webhook en la plataforma 22](#_Toc203357073)

[5. Creación de la tabla en BigQuery 24](#_Toc203357074)

[6. Configuración del Job en Dataflow 26](#_Toc203357075)

[7. Visualización de los datos en BigQuery 28](#_Toc203357076)

[8. Gráficos: 29](#_Toc203357077)

[Conclusión 30](#_Toc203357078)

# Introducción

En el presente informe se describen dos actividades desarrolladas utilizando herramientas del ecosistema de Google Cloud Platform (GCP), con el objetivo de aplicar y comprender distintos procesos de análisis de datos. En la primera actividad, se trabajó con los datos del registro de taxis de la ciudad de Nueva York correspondientes al año 2024, realizando un flujo completo de limpieza, transformación, análisis y visualización, mediante el uso de Cloud Storage, BigQuery, Cloud Dataprep y Looker Studio. En la segunda actividad, se abordó la configuración de un flujo de datos en tiempo real desde un webhook hacia BigQuery, utilizando servicios como Pub/Sub, Cloud Run, Dataflow y BigQuery, lo que permitió construir un sistema escalable, automatizado y con capacidad de análisis casi instantáneo. Ambas experiencias evidencian la potencia y versatilidad de GCP para el manejo de grandes volúmenes de datos y la toma de decisiones basadas en información actualizada.

# Parte 1:

## Arquitectura:

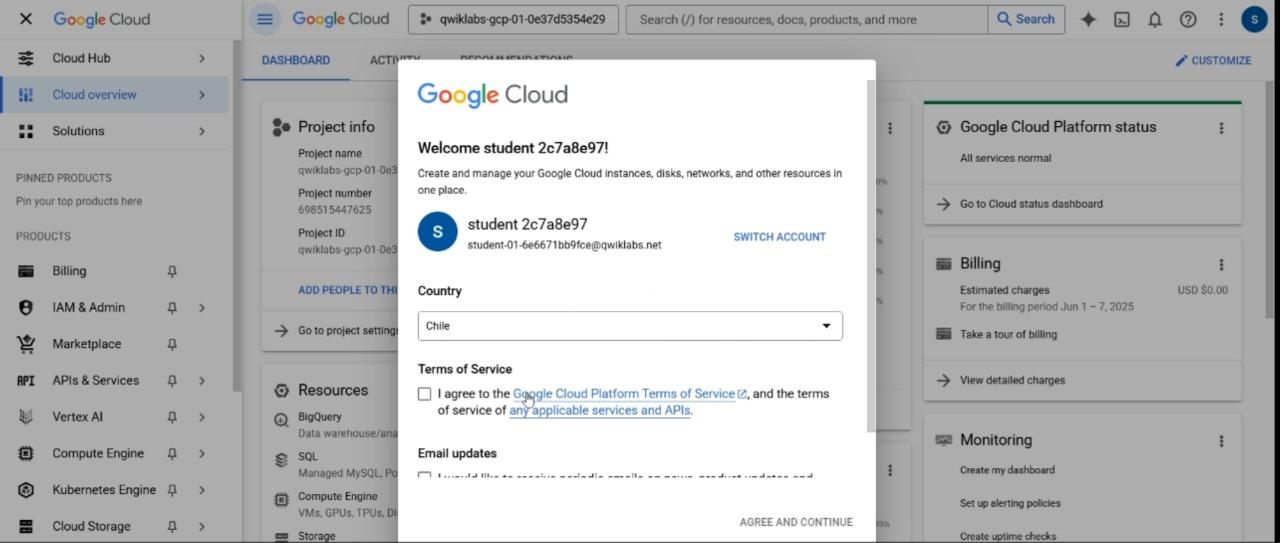
La arquitectura que se muestra representa la idea del desarrollo del problema, ya que, la acumulación de datos de los taxis de NY es grande y con el siguiente modelo, se esperar poder hacer una limpieza de estos datos para obtener beneficios de estos.

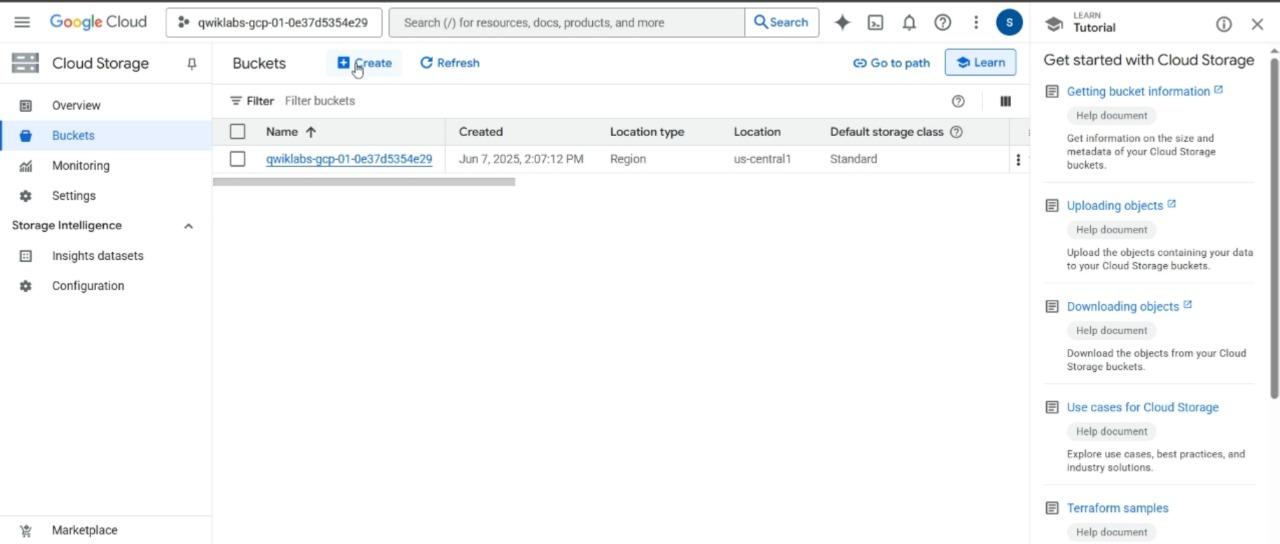
Una captura de pantalla de una computadora

El contenido generado por IA puede ser incorrecto.

## Creación de Buckets:

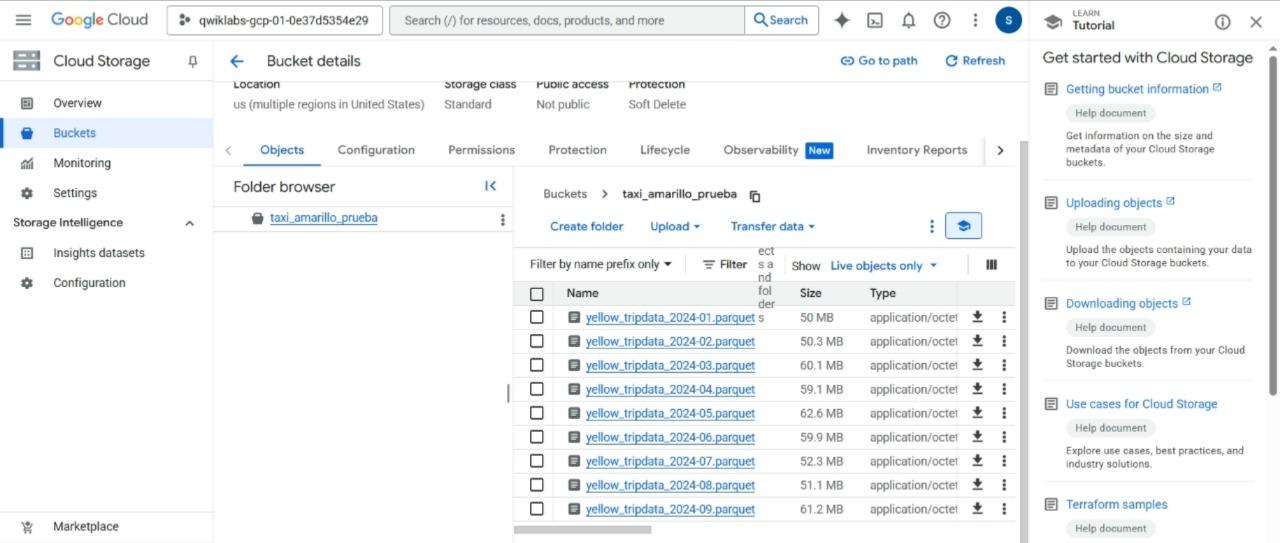
En Google Cloud Plataform (GCP), se espera cargar los datos formato “parquet” que genera la página web de los taxis de NY. Sin embargo, para que estos datos puedan ser cargados, primero procedemos a crear un bucket, donde se cargaran los datos en formato paquet. para luego poder hacer la ingesta de estas en una BigQuery.

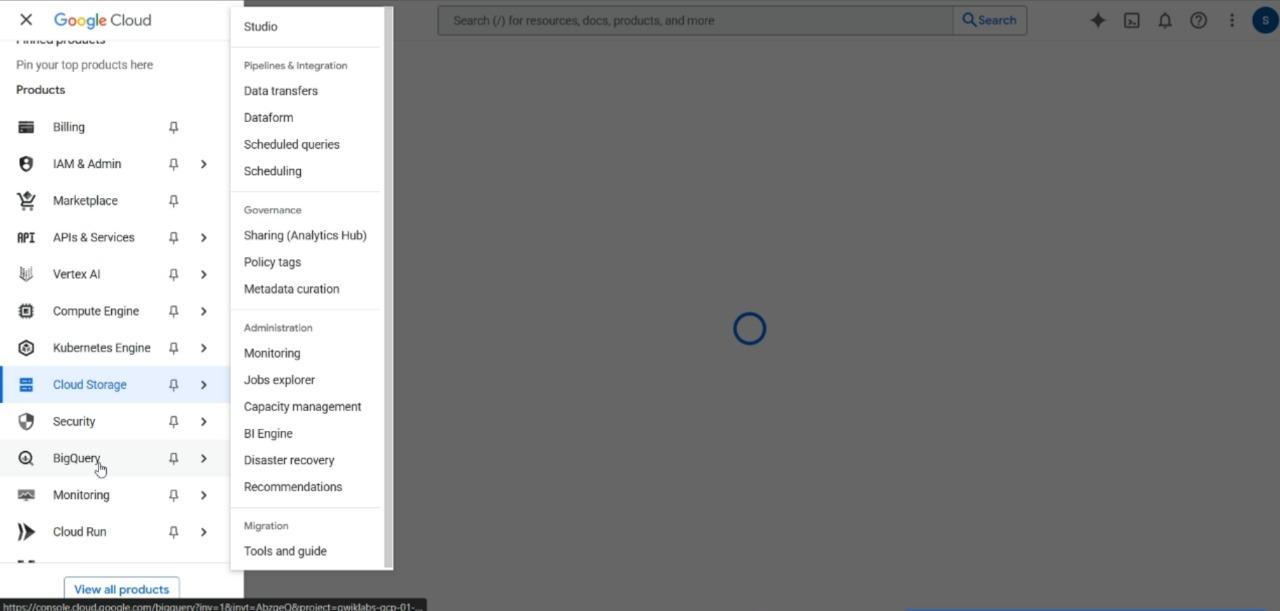




## Carga datos:

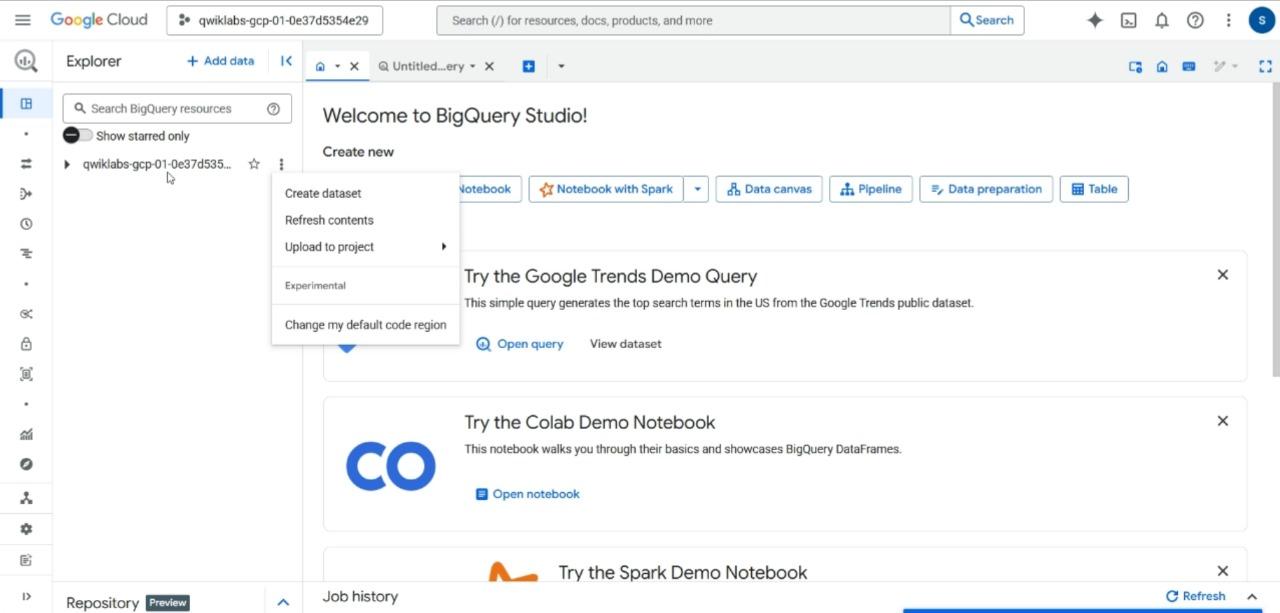
En las siguientes imágenes, se evidencia la carga de datos en el Bucket creado anteriormente en Cloud Storage, se tiene en cuenta que los datos cargados fueron 12 meses del año 2024.

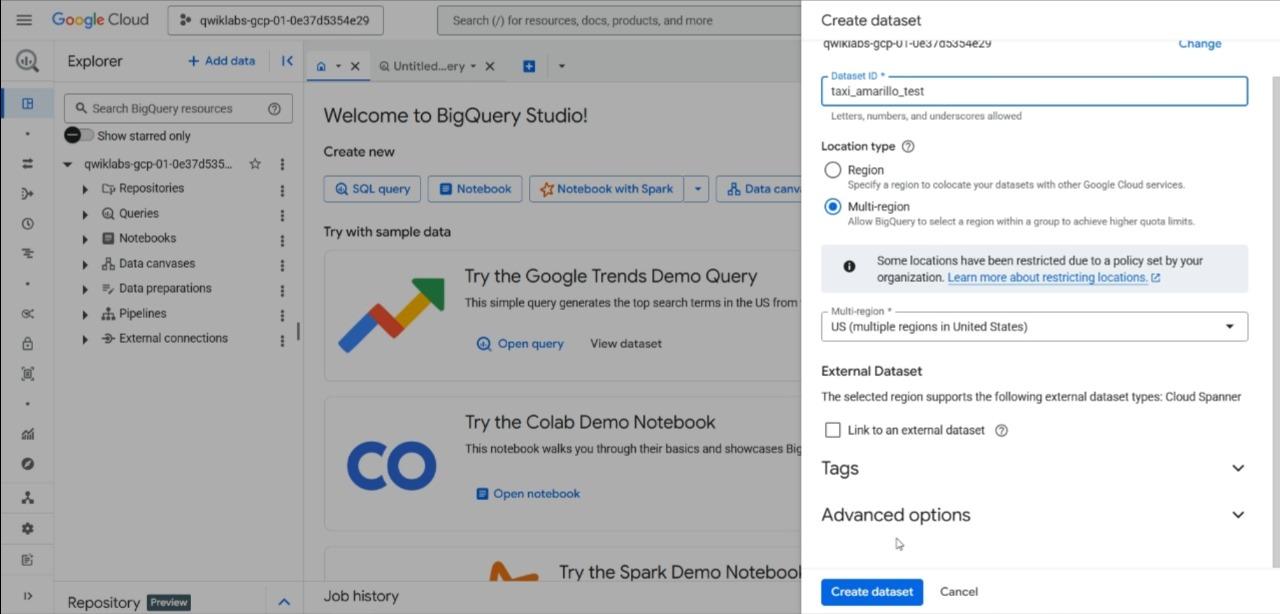




## Creación de BigQuery

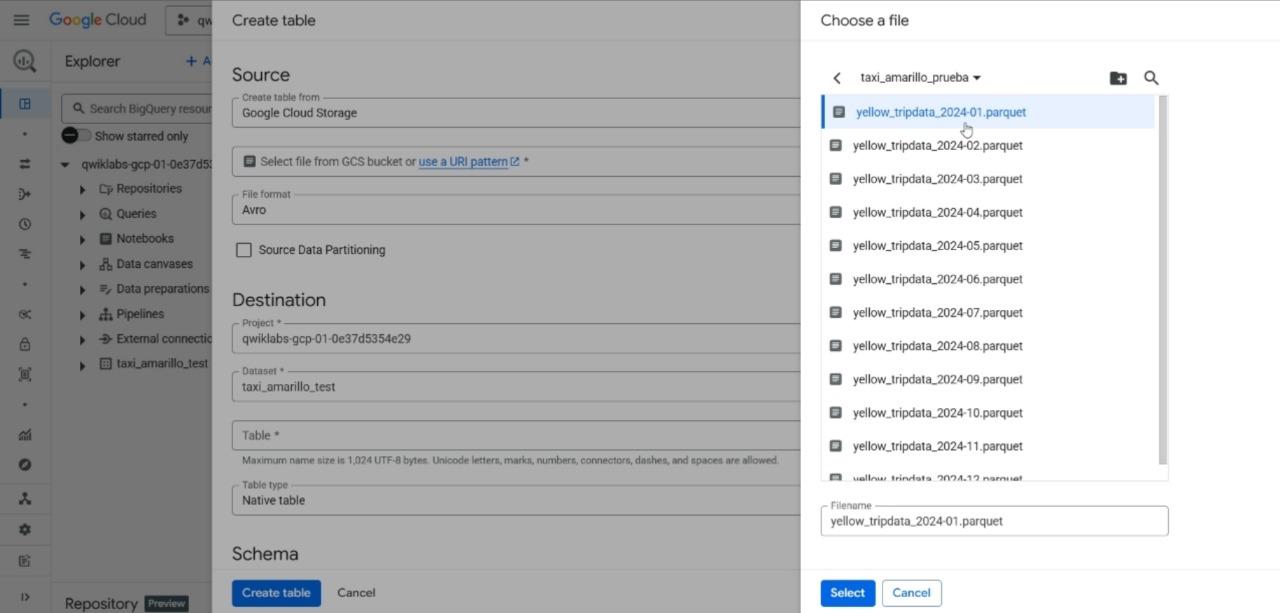
Para que los datos guardados en los Bucket de Cloud Storage puedan ser utilizados, estos se deben cargar en una tabla en BigQuery, para realizar esto, debemos generar un Dataset dentro de nuestro usuario, para luego crear una tabla y conectarla a nuestro Bucket y llamar y mostrar toda la información que contienen los archivos parquet.

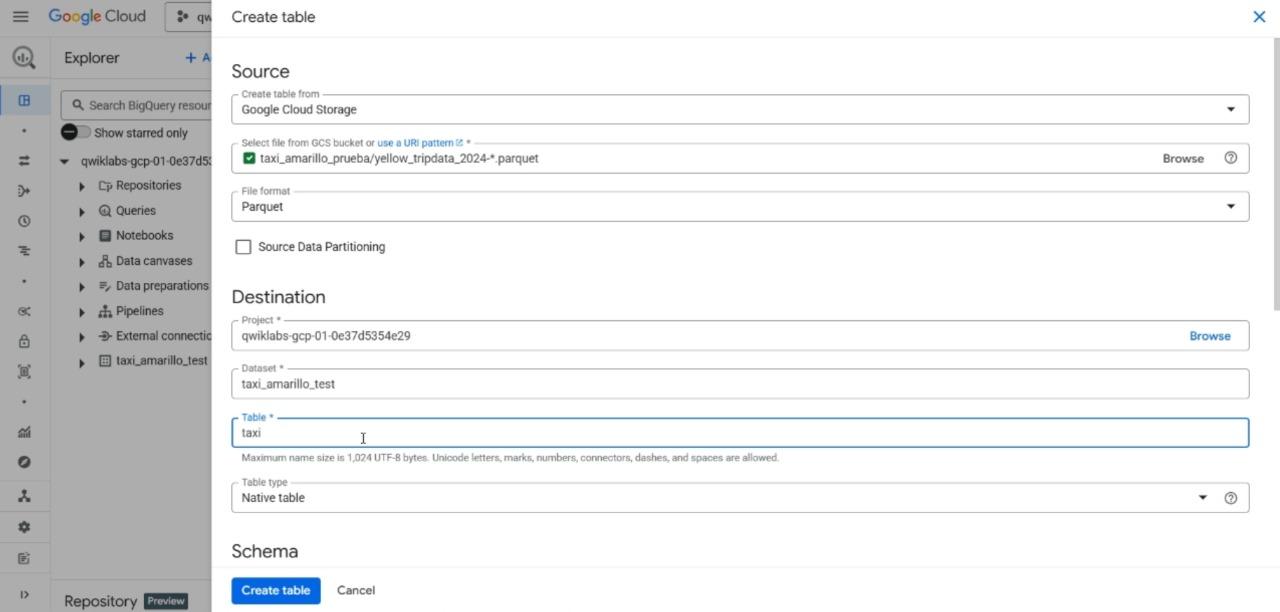




## Creación y carga de parquet

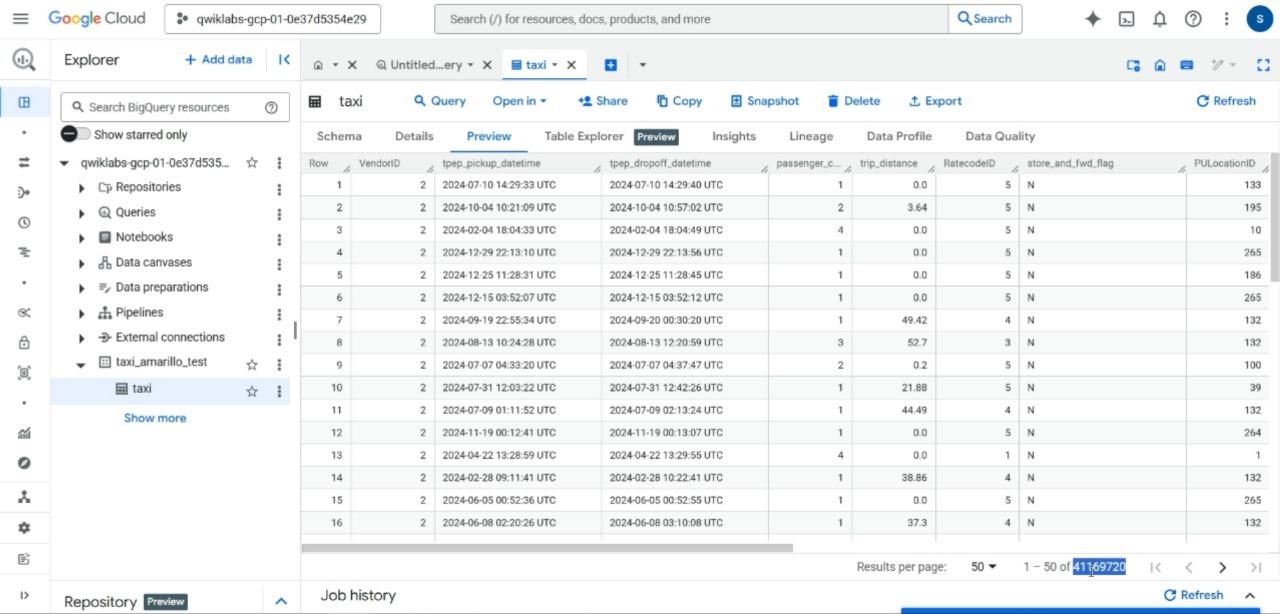
en las siguientes imágenes se evidencia la carga de datos del Bucket en la tabla de la BigQuery.





## Visualizacion

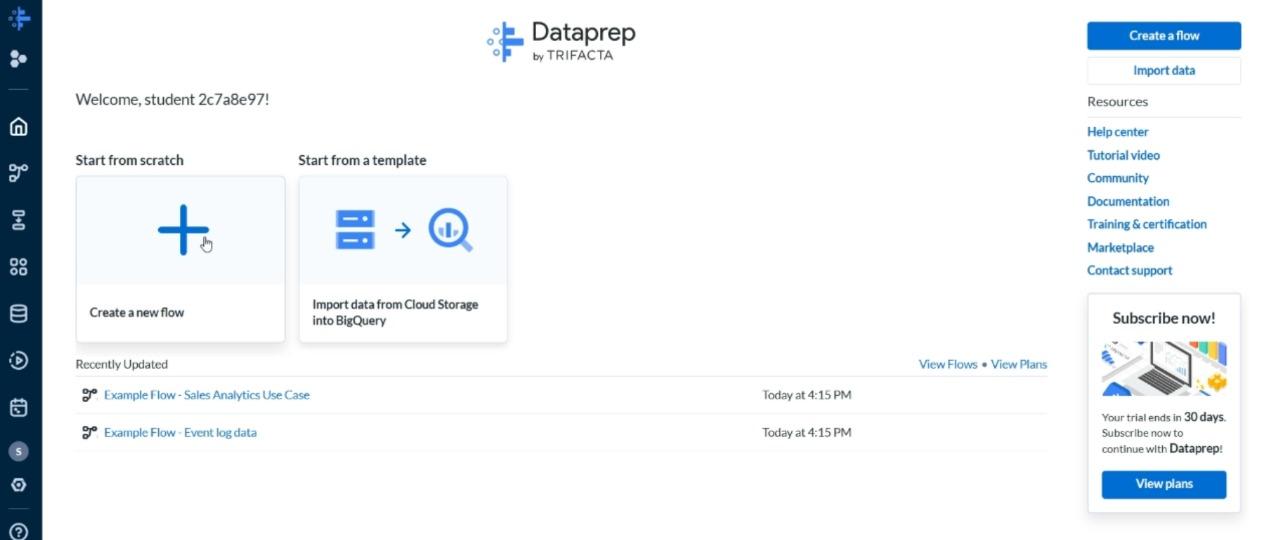
Se realiza una visualización de los datos para comprobar que la carga fue realizada correctamente, esto se hace con una consulta en SQL o con la vista previa de la tabla.



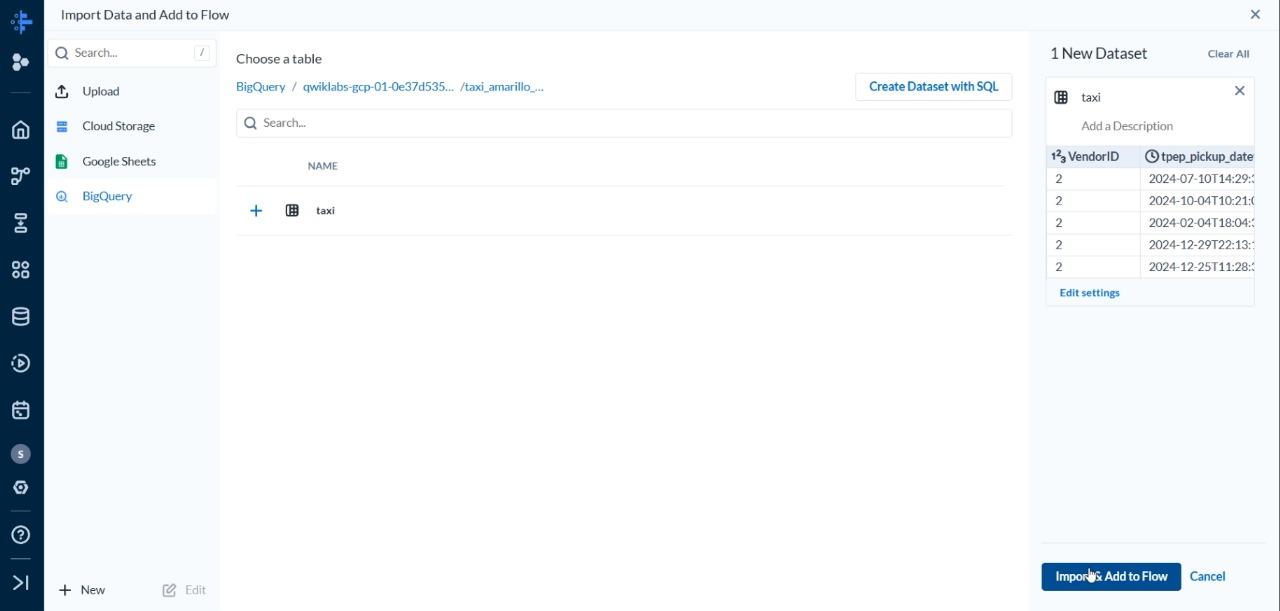
## Limpieza de datos

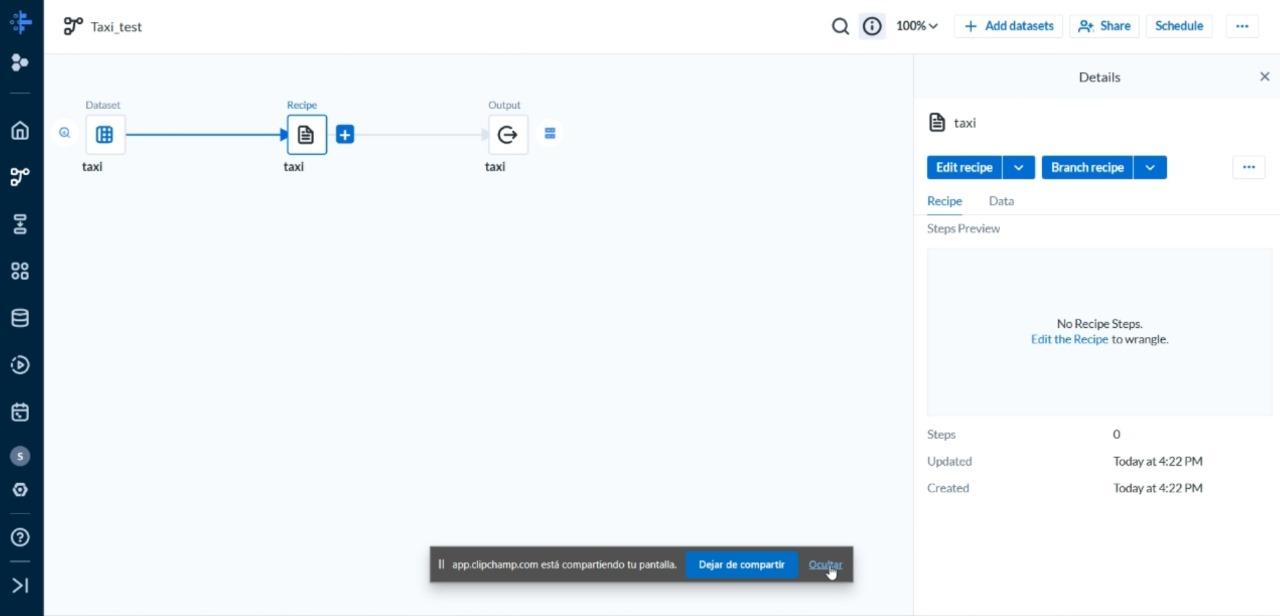
Luego de tener todos los datos cargados, procedemos a conectar los datos de la BigQuery a Dataprep.

Para esto seleccionamos a extencion de gcp, dataprep, para poder crear un flujo de los datos que entran, donde se modifican y luego donde se cargan los datos ya limpios.



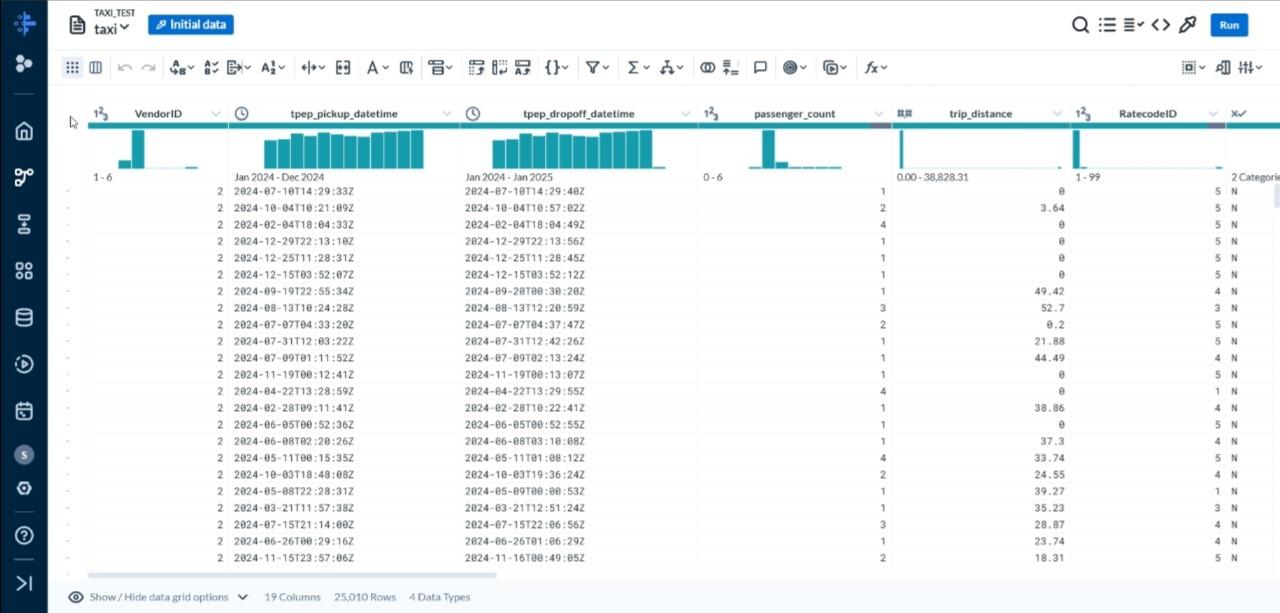
En el siguiente proceso se muestra la conexión de la bigquery con los datos cargados en los pasos anteriores las cuales son incorporadas al flujo para luego tener una visualización de los datos a limpiar.





## Datos:

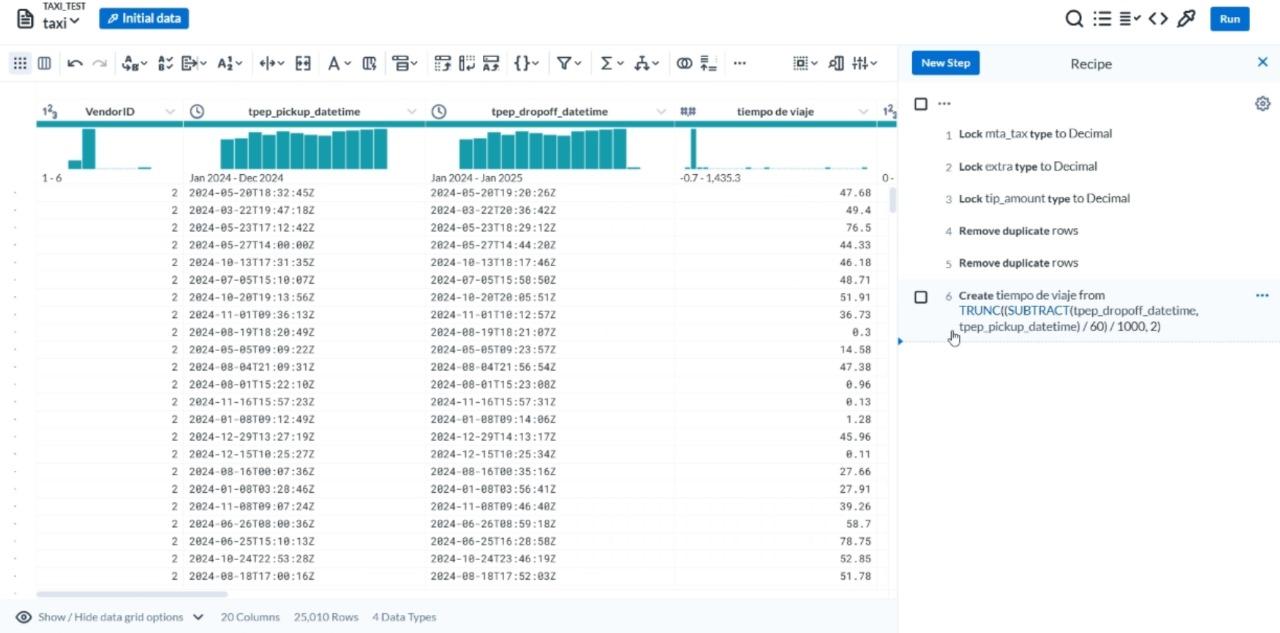
Aquí podes visualizar los datos con sus características las cuales se procederán a limpiar como los datos null, y los datos incompletos.

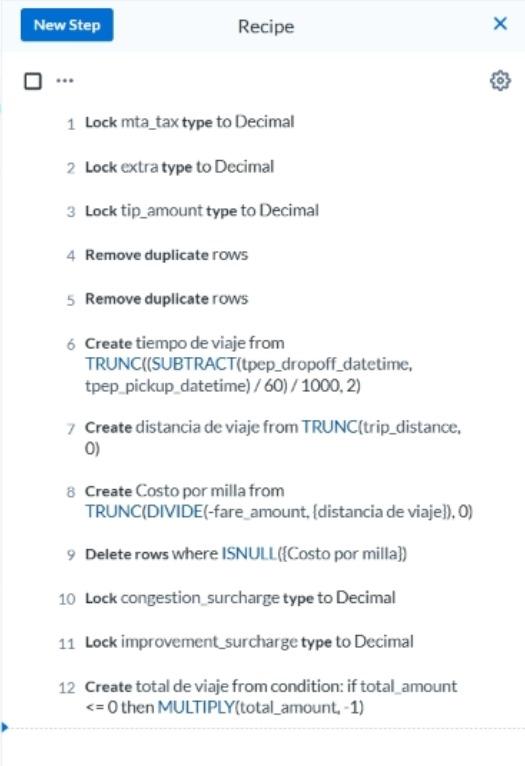
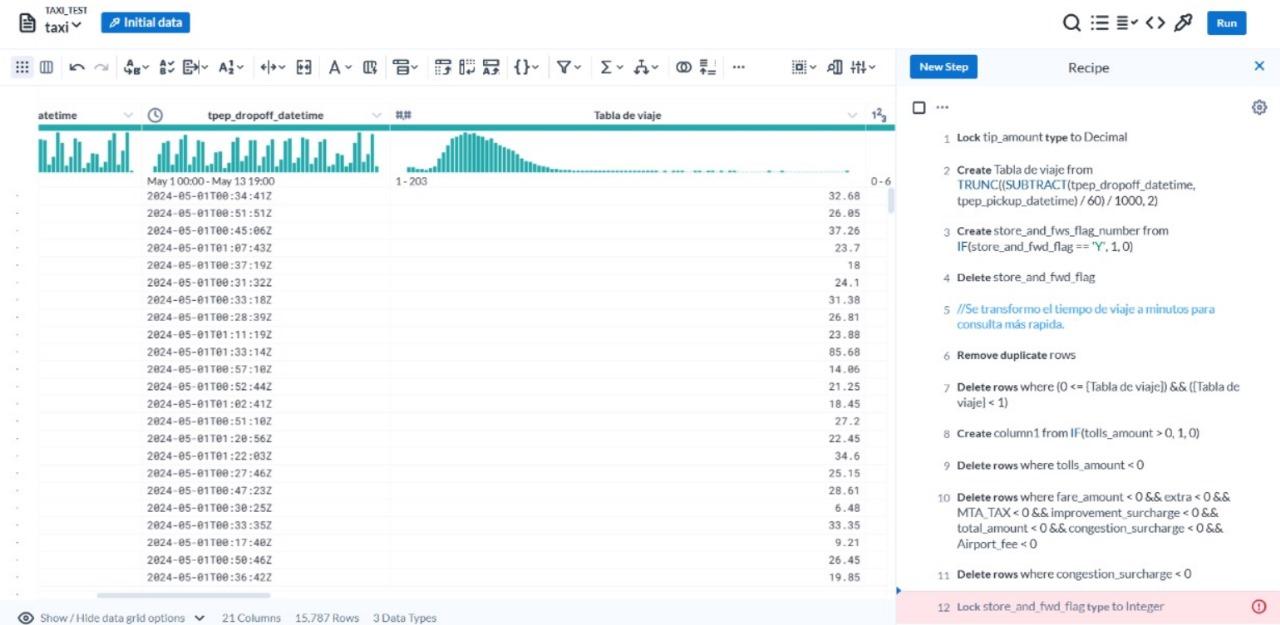




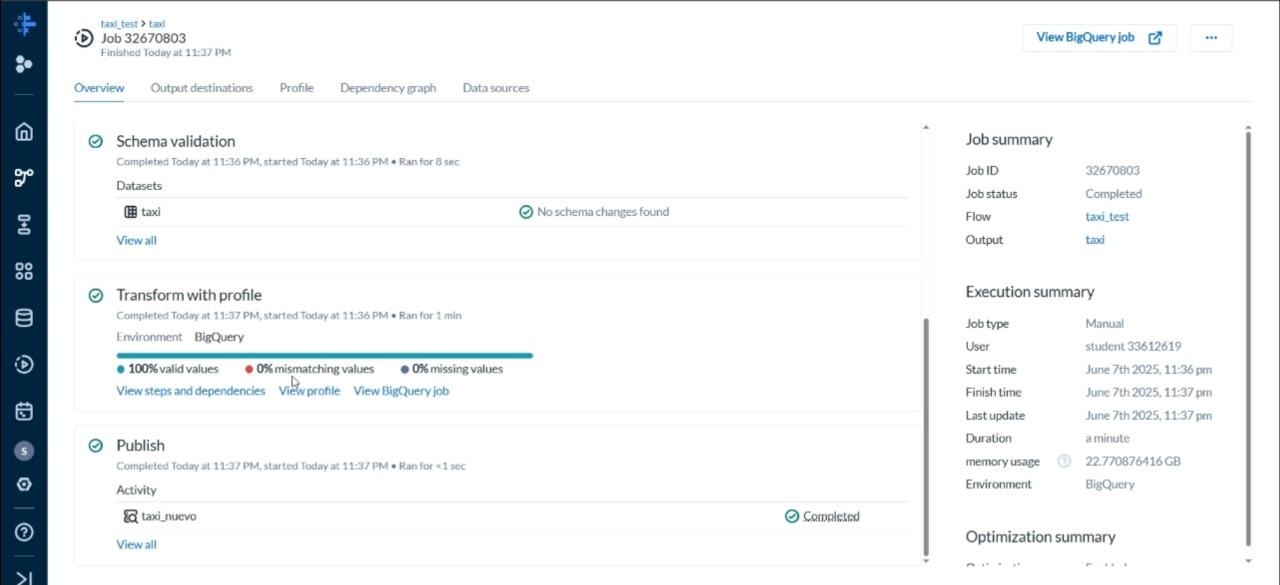
## Creacion de nuevas tablas.

Para un mayor entendimiento se crearon nuevas tablas como “tiempo de Viaje”, “distancia total recorrida” y “costo por milla”. Tambien se eliminaron los datos con poco tiempo de viaje por cosiderarse “viajes no realizados” o “viajes cancelados” los cuales tenían duración de menos de 1 minuto, por ende todos los viajes que tenían tiempo de viaje menor a esta fueron descartados. A su vez, se modifico la tabla de tipo de pago para que fuera de carácter numérica y así contar cuantas personas ocupaban cada tipo de pago para el estudio final.



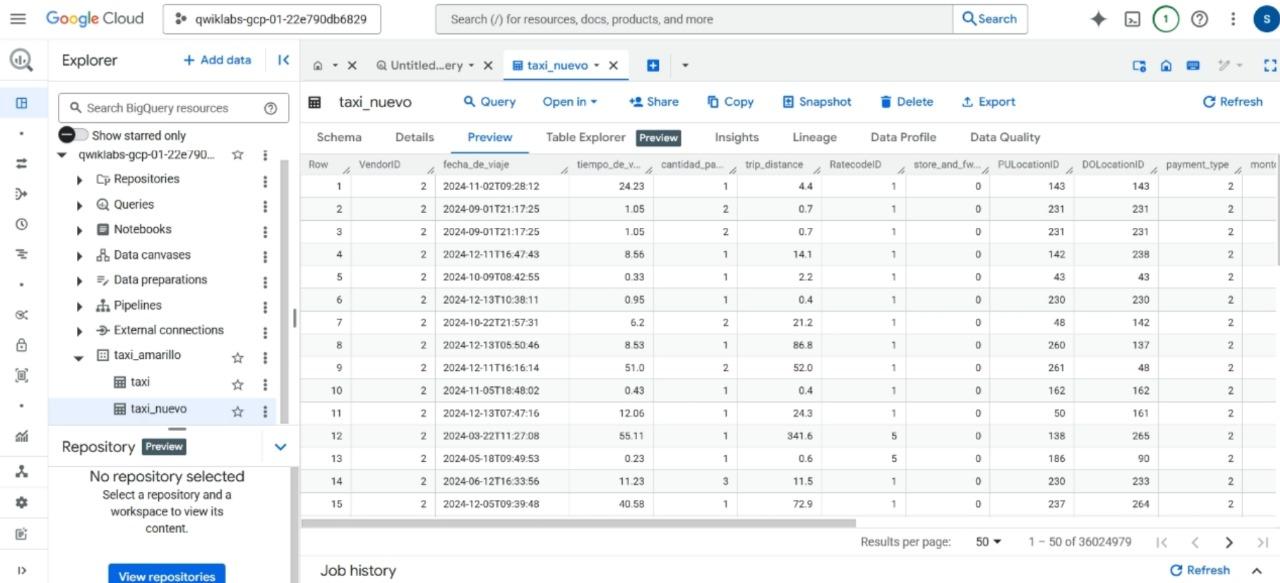


Aquí se evidencia que luego de la carga de datos, el proceso de actualización fue realizado con exito, mostrando coherencia entre los datos que tenían y los nuevos que fueron generados.



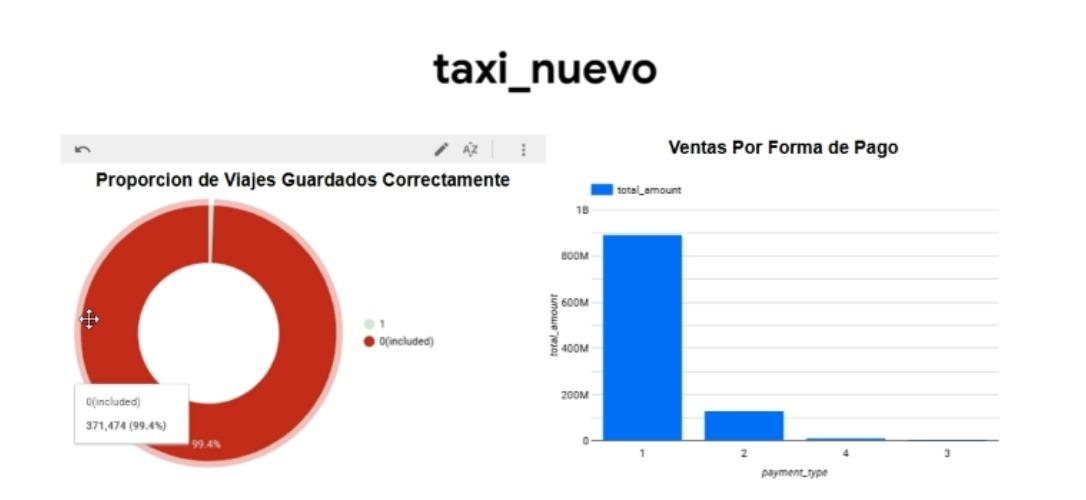
## Comprobar nuevos datos

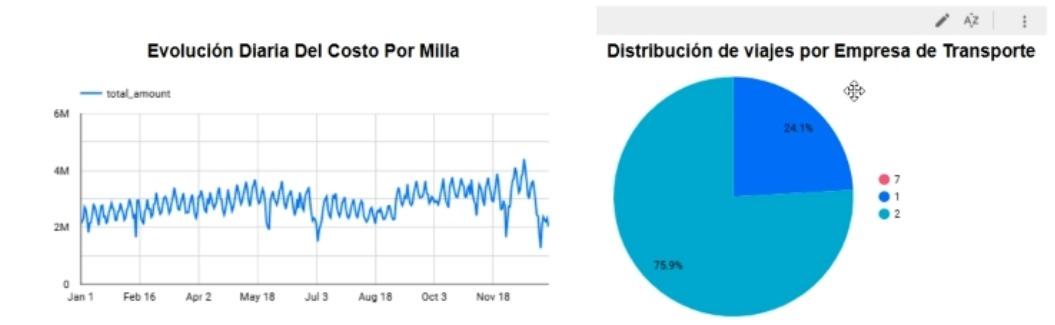
Aquí podemos comprobar por consultas en bigquery que los datos han sido modificado con éxito y están listos para poder ser graficados en Looker Studio, para generar gráficos útiles para la empresa de taxis.



## Gráficos:

Y para finalizar, con los datos ya limpios, podemos crear en Looker Studio visualizaciones graficas de las cifras obtenidas de los parquet de los taxys del año 2024.





# Parte 2:

## Arquitectura:

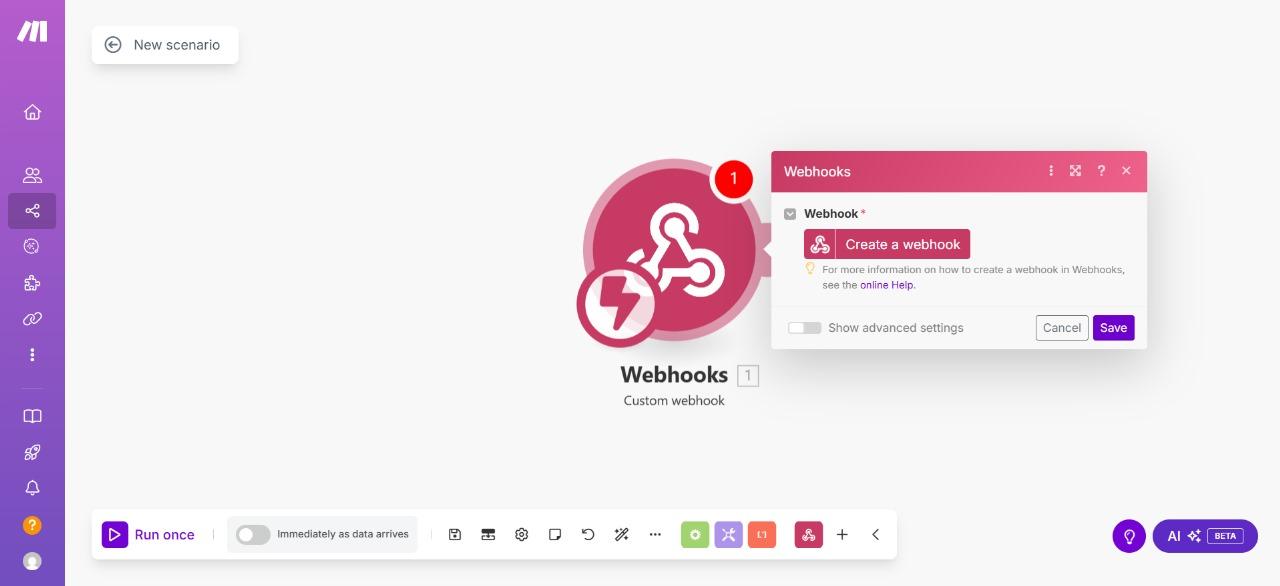
La arquitectura representada, pretende resolver el problema de procesamiento de datos via streaming (real time), y que es necesario generar limpieza de datos para evitar duplicados o datos erróneos.

Una captura de pantalla de una computadora

El contenido generado por IA puede ser incorrecto.

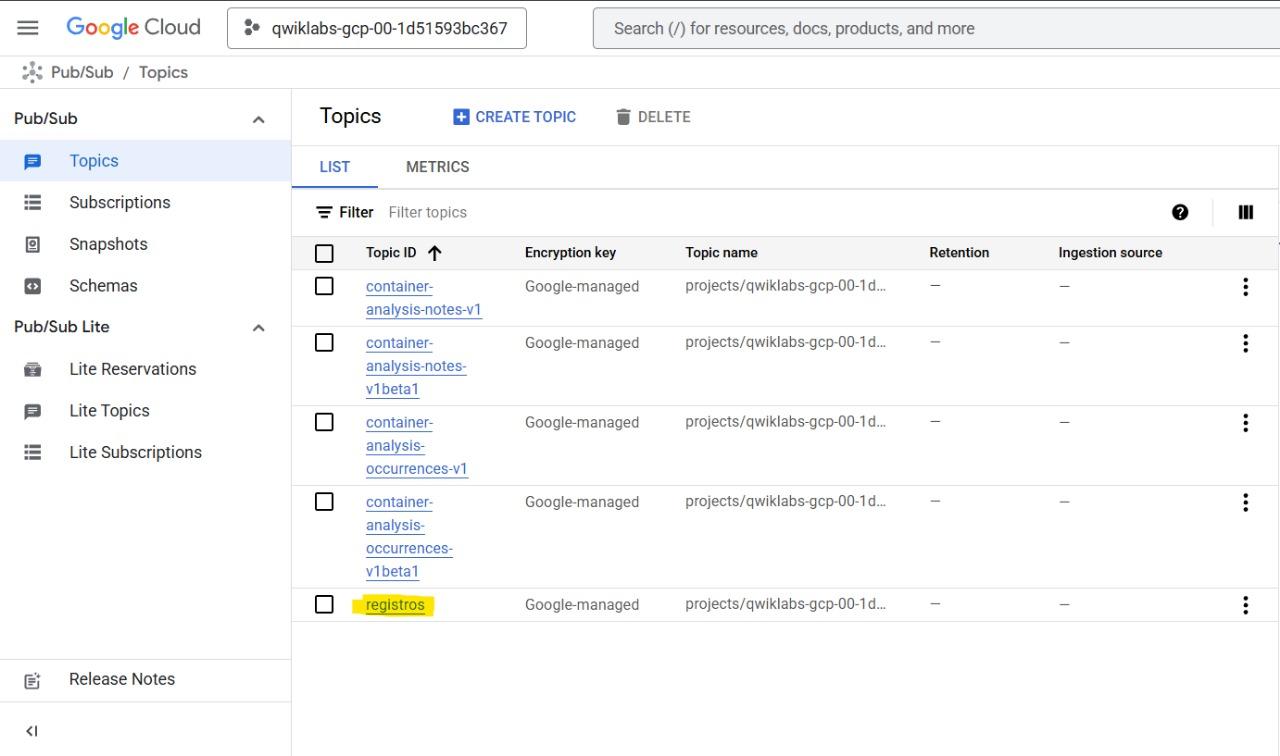
# Creación del Webhook

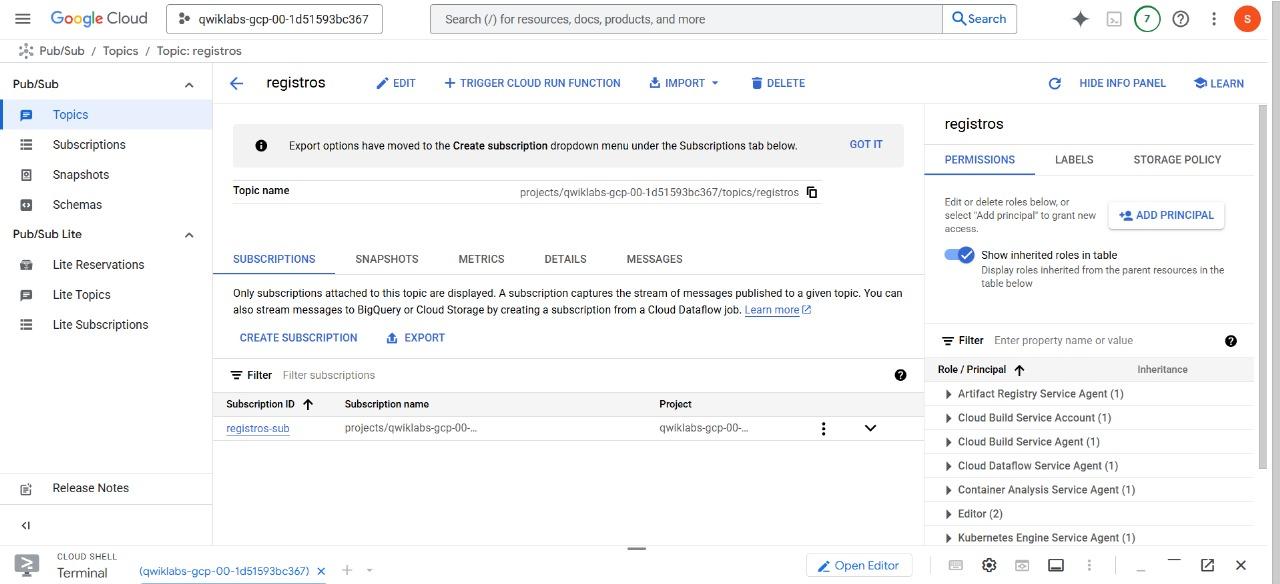
Se utilizó la plataforma Make.com para crear un webhook personalizado capaz de recibir datos JSON. Esta URL sirve como punto de entrada para los datos en tiempo real junto con la estructura de los datos que serán procesados más adelante.



# Configuración del tópico en Pub/Sub

Dentro de Google Cloud, se crea un “tópico” llamado registros que actuará como receptor de los datos entrantes. Este tópico almacena los mensajes que luego serán redirigidos hacia otros servicios, como BigQuery.



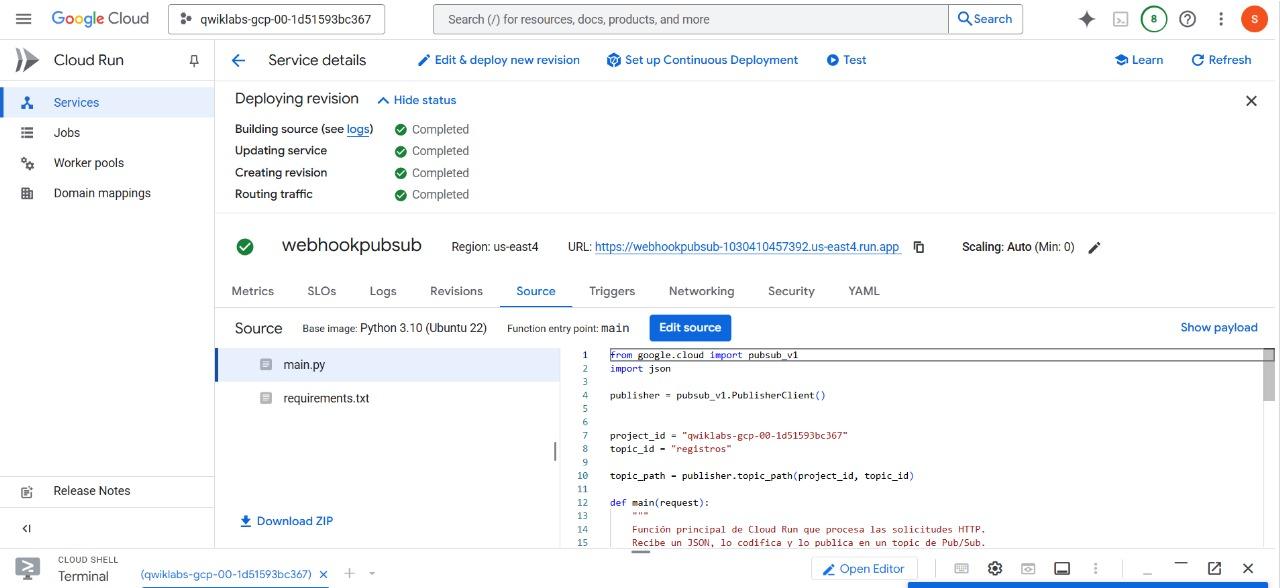


# Desarrollo y despliegue en Cloud Run

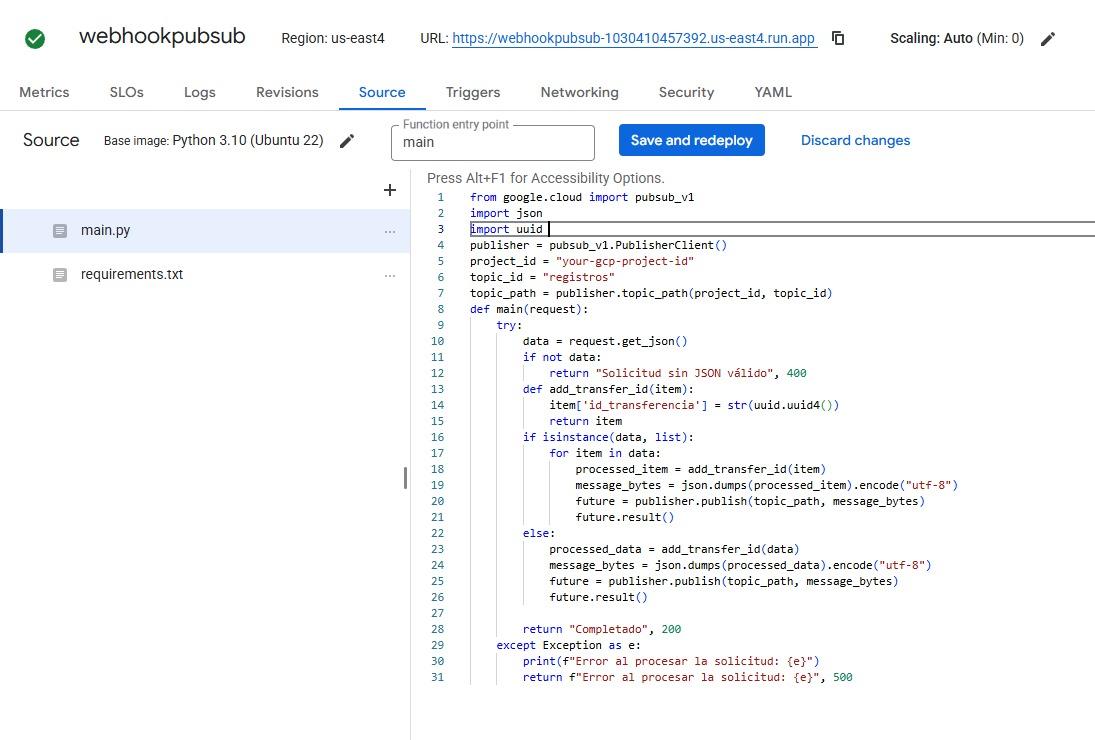
Se configura una función en Cloud Run para recibir los datos del webhook y enviarlos al tópico registros. Esta función actúa como un intermediario inteligente entre el Webhook y Pub/Sub.

La configuración incluye:

* Nombre y región
* Lenguaje de programación
* Permisos de acceso no autenticado
* Despliegue del servicio y obtención de la URL pública







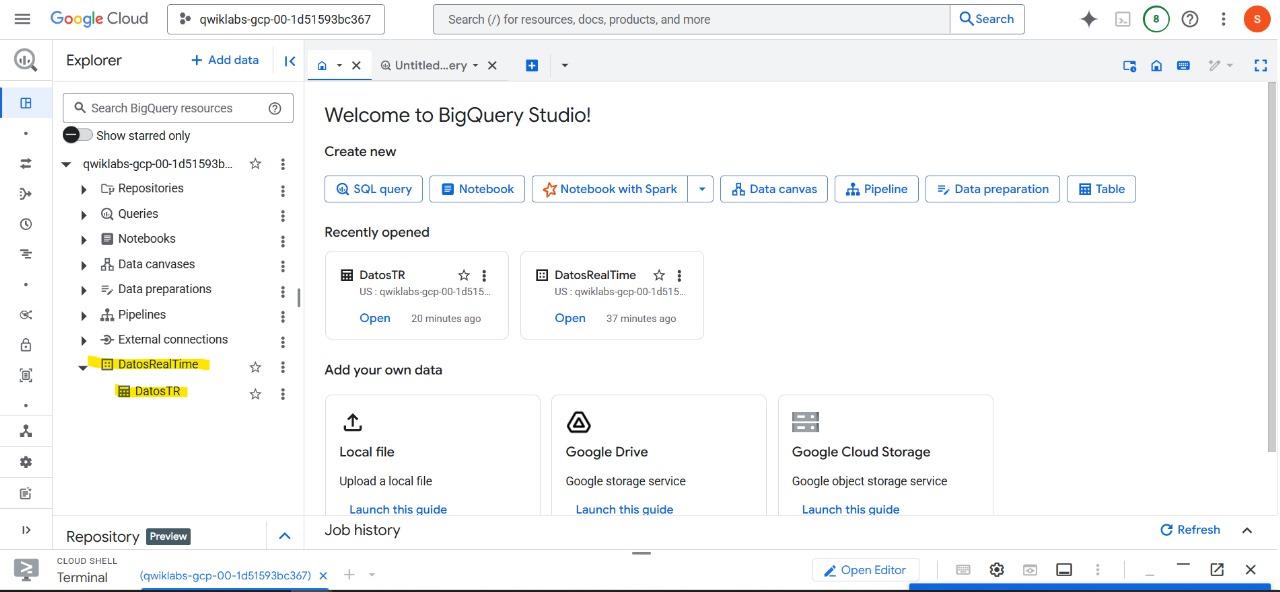
# Registro del Webhook en la plataforma

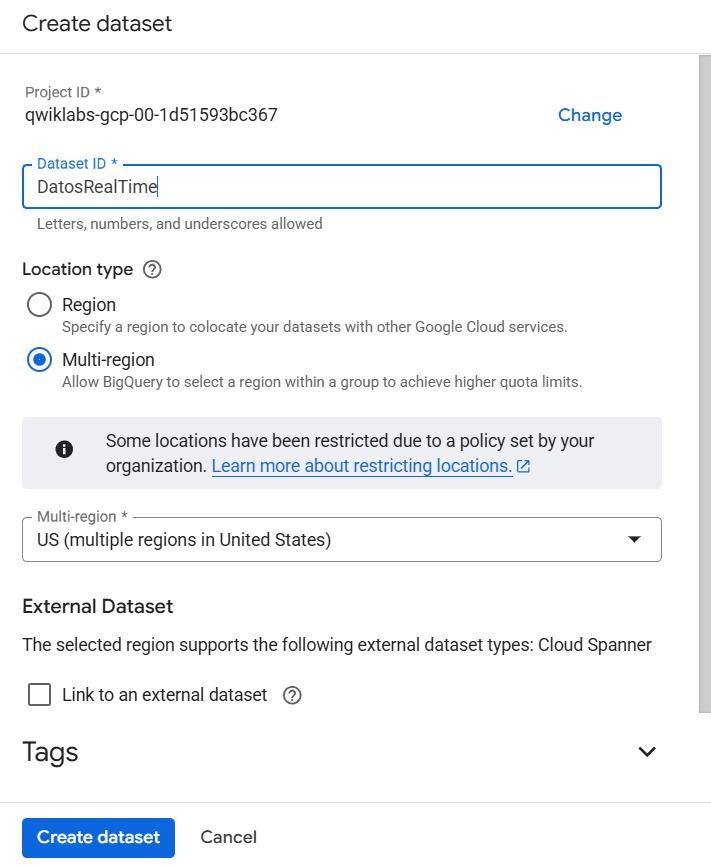
La URL obtenida desde Cloud Run es registrada en el portal <https://bdrealtimeescuelait.duoc.cl>. Esto permite que los datos enviados desde el navegador o simulador se redirijan directamente a la función desplegada en la nube.

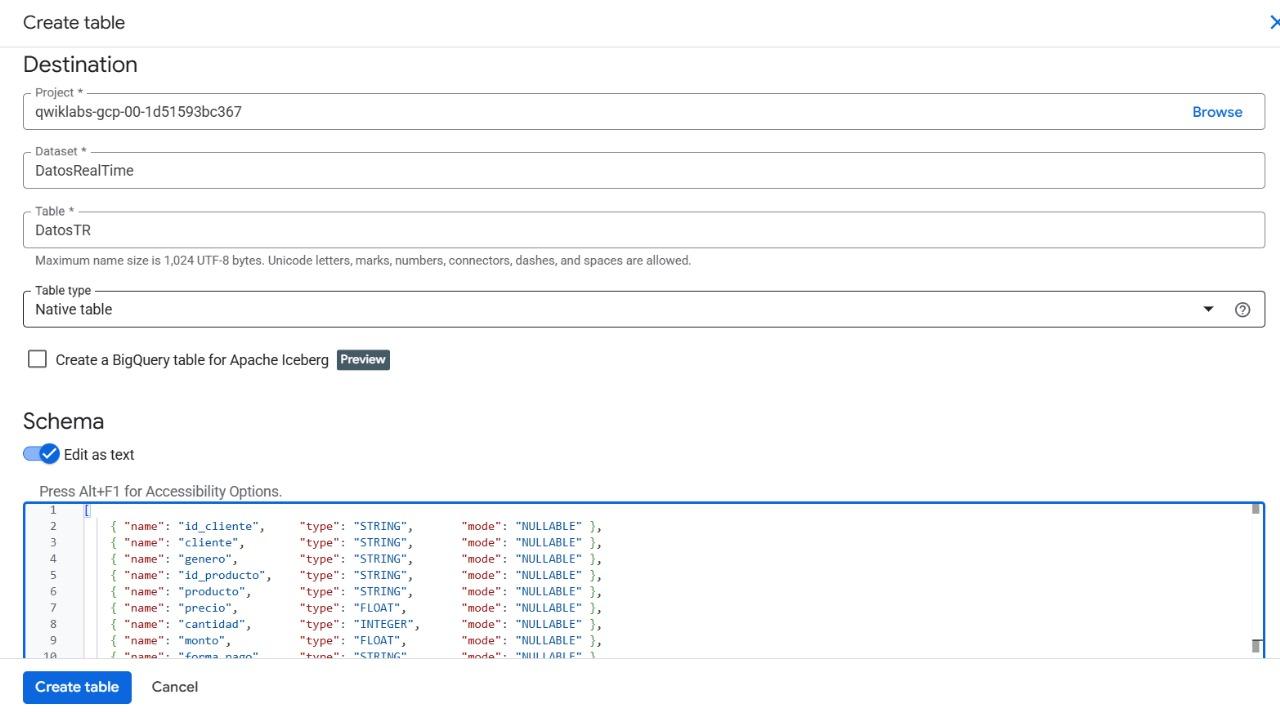


# Creación de la tabla en BigQuery

En el servicio de BigQuery se crea un conjunto de datos llamado DatosRealTime y dentro de él, una tabla llamada DatosTR, con campos definidos para almacenar datos como: cliente, producto, precio, fecha, etc.







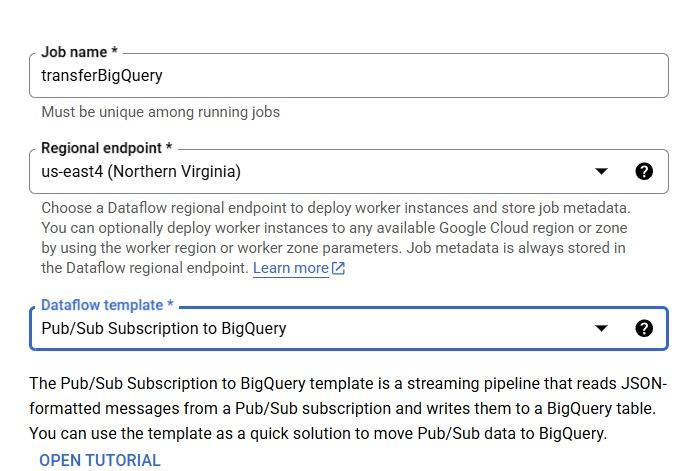
# Configuración del Job en Dataflow

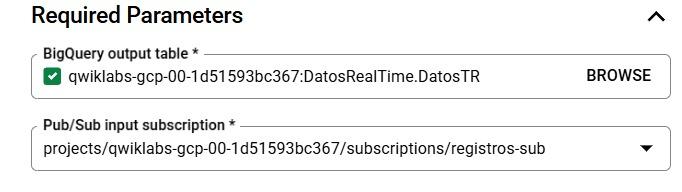
Dataflow permite transferir los datos desde el tópico de Pub/Sub hacia BigQuery. Se utiliza una plantilla predeterminada llamada “Pub/Sub Subscription to BigQuery”. En esta etapa se configuran:

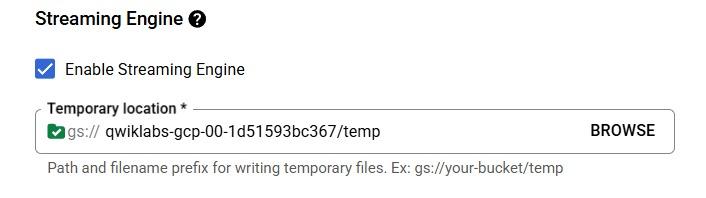
La suscripción al tópico registros

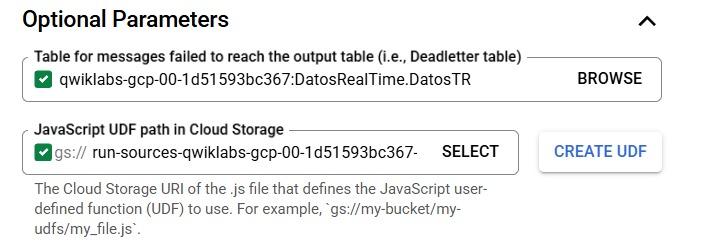
La tabla de destino en BigQuery

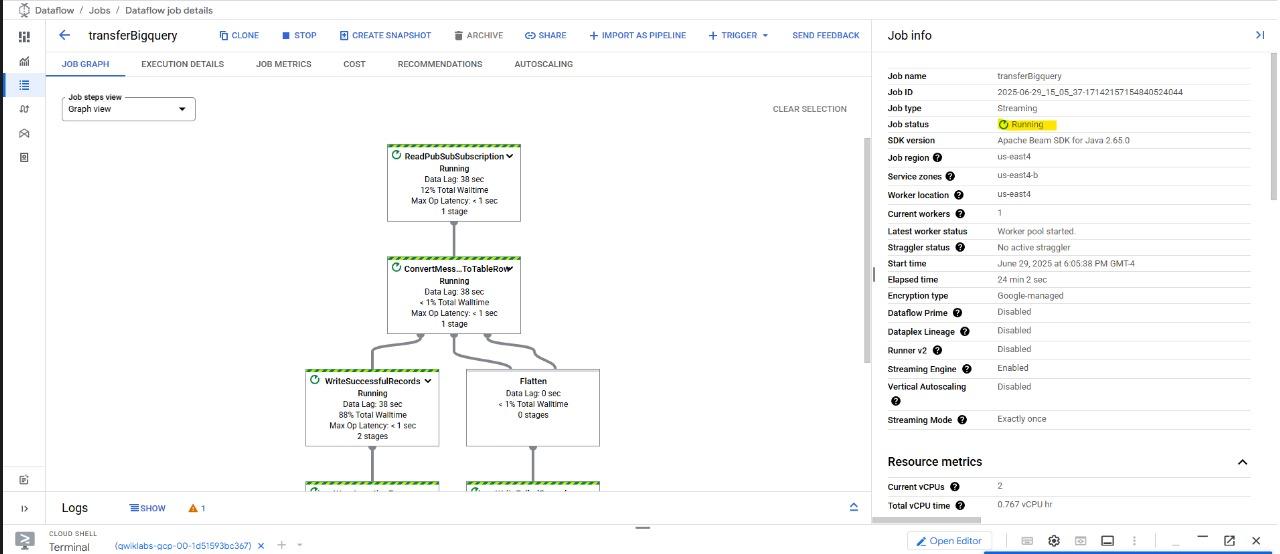
Un bucket temporal para el procesamiento





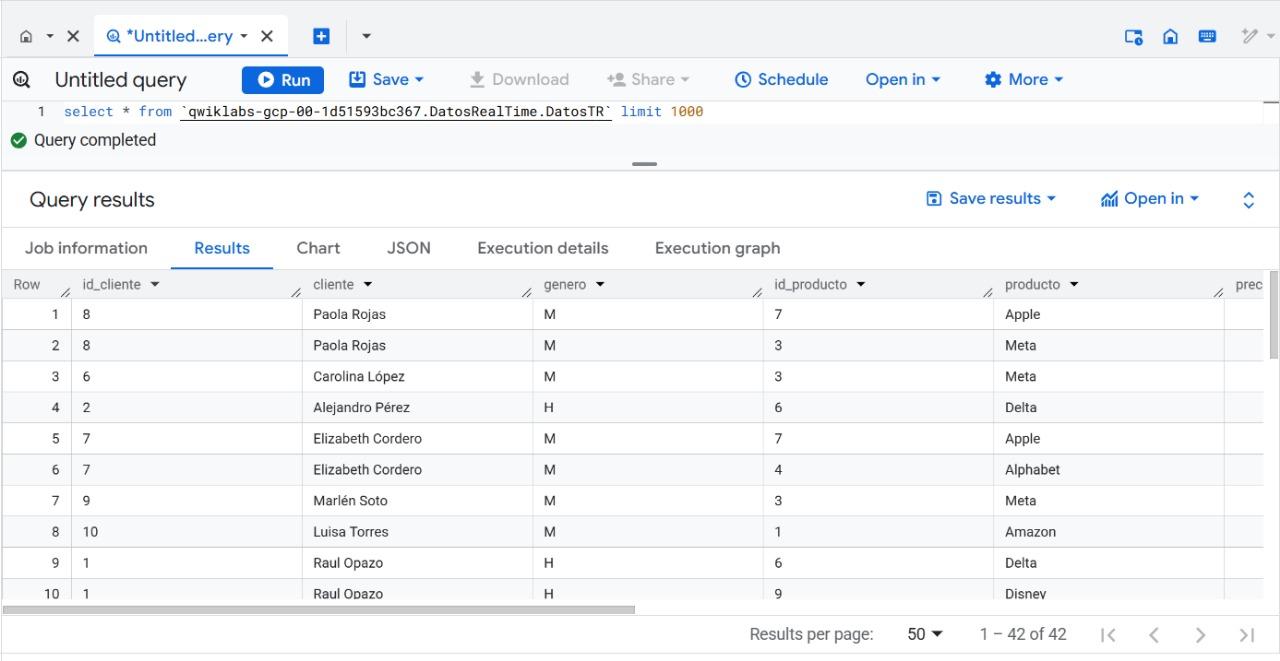




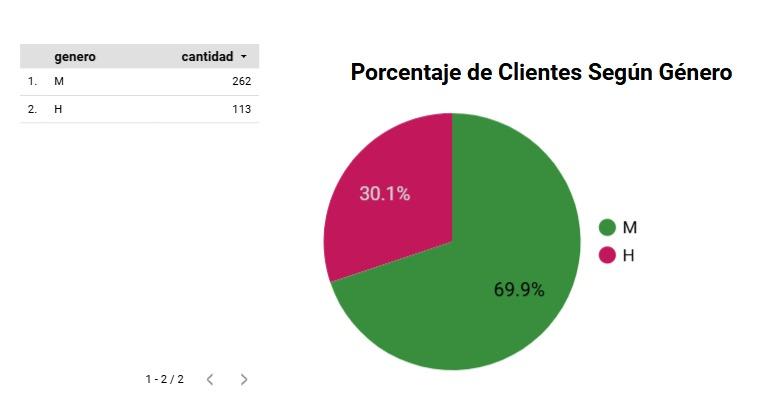


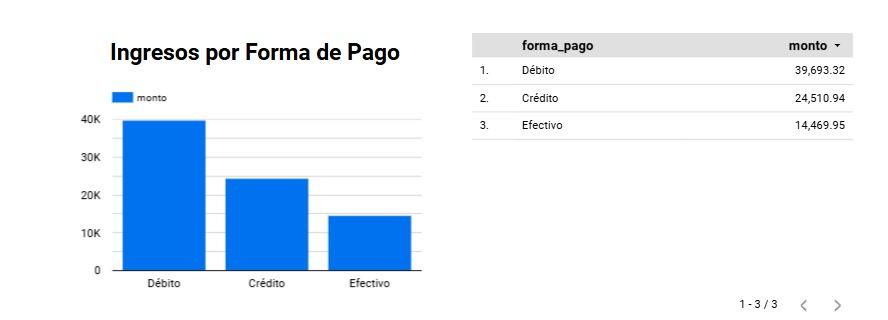
# Visualización de los datos en BigQuery

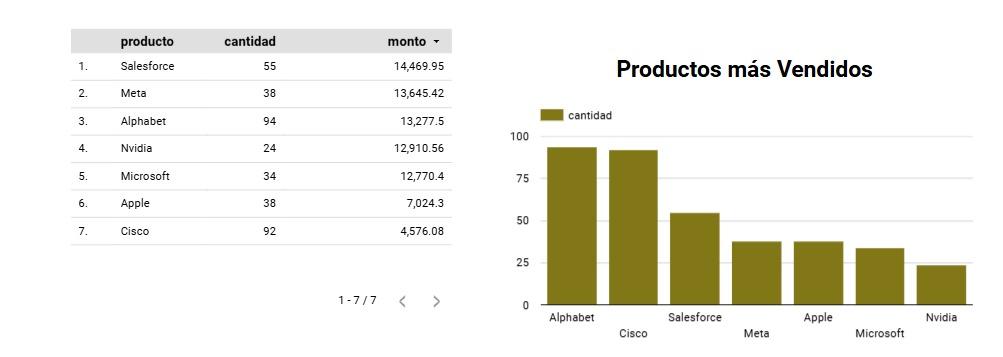
Una vez ejecutado el Job, los datos comienzan a almacenarse automáticamente en la tabla DatosTR. Desde la interfaz de BigQuery se pueden realizar consultas para explorar los registros en tiempo real.



# Gráficos:







# Conclusión

A lo largo de este proyecto se pudo comprobar que Google Cloud Platform ofrece un entorno robusto y eficiente para trabajar con grandes volúmenes de datos. El uso de herramientas como Cloud Storage, BigQuery, Dataprep y Looker Studio permitió realizar un proceso completo de recolección, limpieza, procesamiento, análisis y visualización de los datos de manera automatizada y escalable. Esta integración facilitó el manejo ordenado y ágil de la información, permitiendo obtener resultados claros en menos tiempo. En consecuencia, se evidencia que contar con una infraestructura en la nube bien estructurada como la de Google Cloud no solo simplifica el trabajo técnico, sino que también mejora significativamente la toma de decisiones basadas en datos actualizados y confiables.