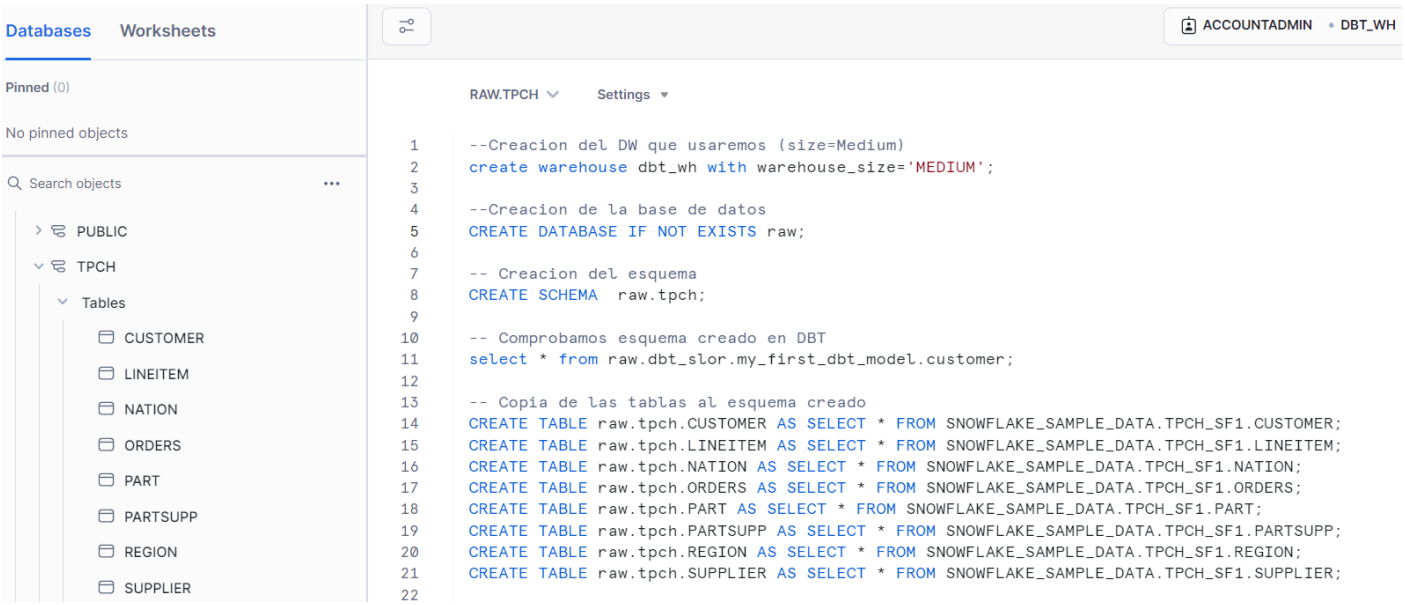
Entrega Final Proyecto

El primer paso a realizar fue la conexión entre las distintas plataformas para la gestión del proyecto. Por un lado, Snowflake , es un almacén de datos en la nube que permite el almacenamiento y análisis de grandes volúmenes de datos. DBT es una herramienta de transformación de datos basada en SQL y que tomará gran importancia durante el proyecto y GitHub es una plataforma de desarrollo colaborativo que ofrece control de versiones para proyectos de software.

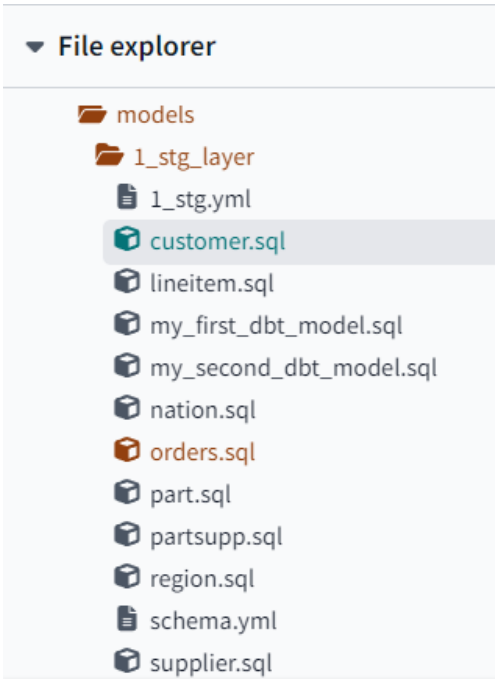
SNOWFLAKE

A continuación, se realiza un análisis superficial para saber qué información contienen las tablas y que significa cada campo. En esta primera fase, copio las tablas a otro esquema en snowflake ya que el original no es editable.



DBT

-Creo una carpeta en DBT, *'stg_layer'*, para proceder a la ingestión de los datos y a realizar pequeños cambios como la eliminación de la columna *'comment'* que no proporciona información de utilidad o el uso de otros nombres para ciertas columnas que son ambiguas.



-Ejemplo:

models > 1_stg_layer > customer.sql

```sql
With customer AS (
    SELECT
        c_custkey,
        c_name as customer_name,
        c_address as customer_address,
        c_nationkey,
        c_acctbal,C_MKTSEGMENT
    FROM tpch.customer
)

Select * from customer
```

-A continuación, para la segunda capa del proyecto en la que aplicaremos algunas transformaciones más notorias,decido crear la carpeta *'mid_layer'*. En la que creo distintas métricas que pueden ser de utilidad para posteriores análisis. Algunas de ellas son:

-precio descontado por artículo en una línea de pedido
-precio base por artículo en una línea de pedido
-Cantidad total del descuento (valor producto-menos el descuento-menos el impuesto) por artículo

Podrían servir para responder a preguntas como:

-Como varía el precio descontado por articulo en una línea de pedido según el tipo de cliente o segmento de mercado.
-Porcentaje del precio final de venta que se atribuye al descuento y al impuesto por artículo en una línea de pedido.
-Influencia del descuento e impuesto en la compra de ciertos artículos

mid_orders_line.sql

models > 2_mid_layer > mid_orders_line.sql

```sql
with mid_orders_line AS (
    SELECT
        li.L_ORDERKEY,
        o.O_ORDERKEY,
        o.O_CUSTKEY,
        o.O_ORDERDATE,
        li.L_LINESTATUS AS order_status_code,
        li.L_PARTKEY,
        li.L_SUPPKEY,
        li.L_RETURNFLAG,
        li.L_LINENUMBER,
        li.L_SHIPDATE,
        li.L_COMMITDATE,
        li.L_RECEIPTDATE,
        li.L_SHIPMODE,
        li.L_EXTENDEDPRICE / NULLIF(li.L_QUANTITY, 0) AS item_price, --precio base por artículo en una línea de pedido
        li.L_DISCOUNT,
        (li.L_EXTENDEDPRICE * (1 - li.L_DISCOUNT)) AS item_discounted_price, -- precio descontado por artículo en una línea de pedido
        li.L_EXTENDEDPRICE,-- precio total articulo sin descuentos
        li.L_TAX,--tasa impuestos artículo
        ((li.L_EXTENDEDPRICE + (-1 * li.L_EXTENDEDPRICE * li.L_DISCOUNT)) * li.L_TAX) AS item_tax_amount -- cantidad total del impuesto calcula
    FROM
        tpch.lineitem li
    JOIN
        tpch.orders o ON o.O_ORDERKEY = li.L_ORDERKEY
    ORDER BY
        o.O_ORDERDATE
```

Defer to production

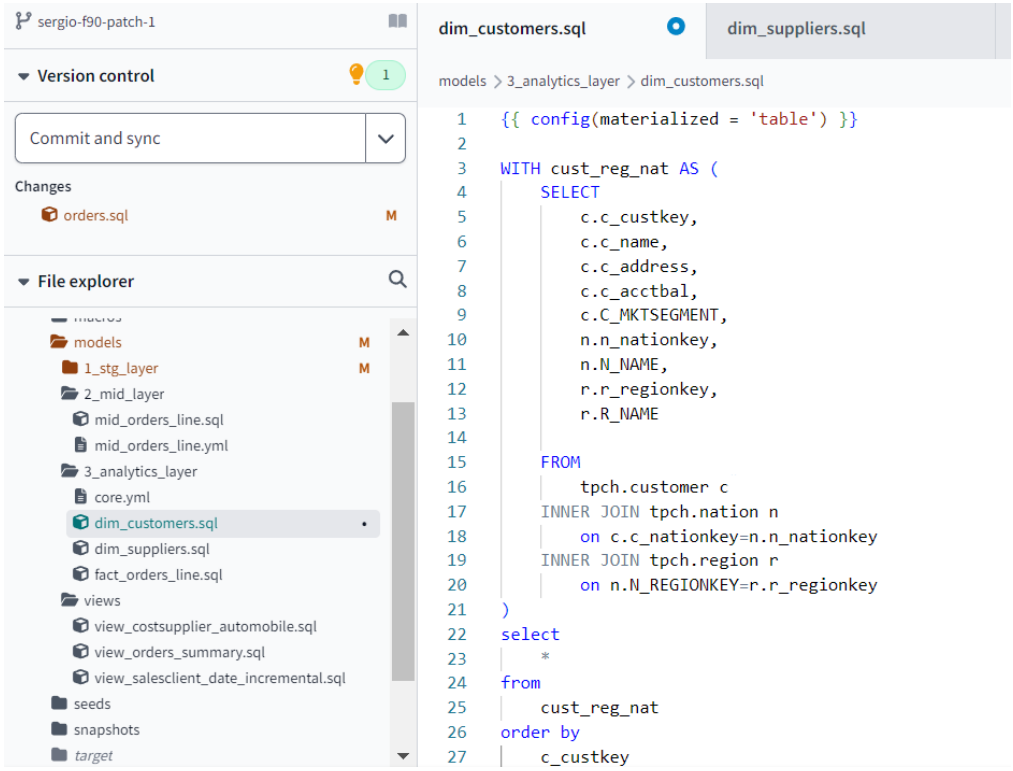
-Para la última capa, la cual llamo *'analytics_layer'*, defino la tabla de hechos a partir de la *'mid_layer'* seleccionando distintas operaciones,como la media o el sumatorio de distintas métricas anteriores.

fact_orders_line.sql

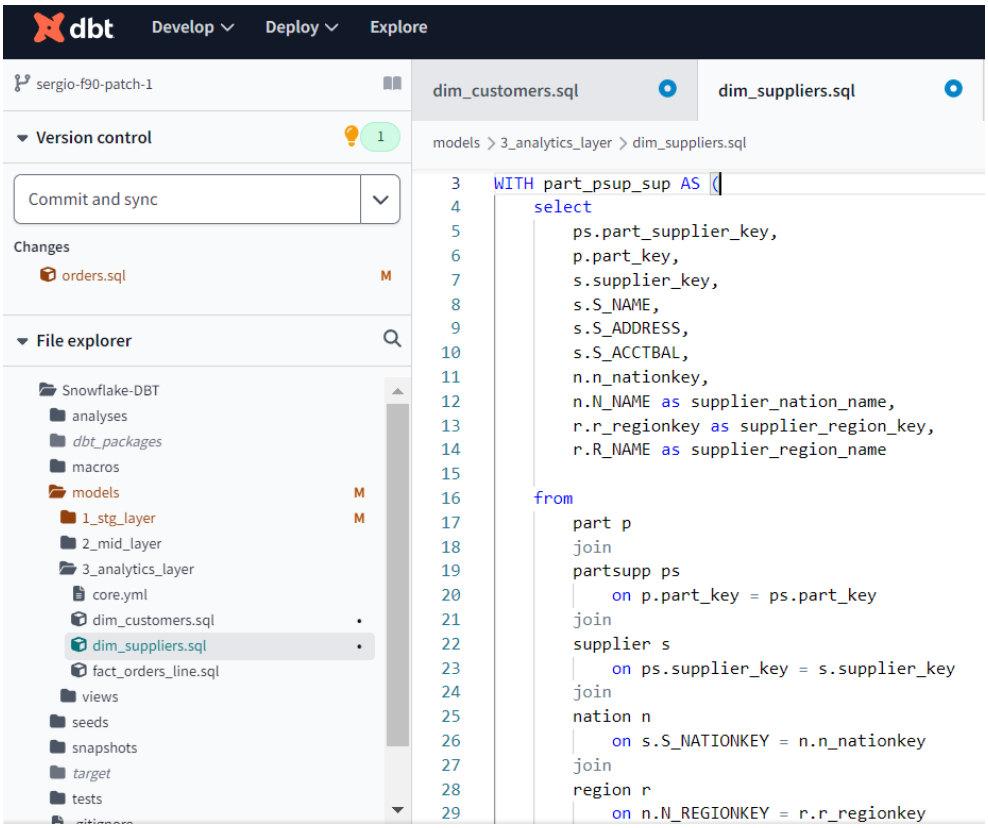
models > 3_analytics_layer > fact_orders_line.sql

```sql
WITH orders_line_agg AS (
    SELECT
        ol.O_ORDERKEY,
        ol.item_discounted_price as discounted_price_item,
        AVG(ol.item_discounted_price) AS avg_discounted_price,
        SUM(ol.item_discounted_price) AS item_discounted_amount,
        SUM(ol.item_tax_amount) AS item_tax_amount
    FROM
        mid_orders_line ol
    GROUP BY
        ol.O_ORDERKEY,discounted_price_item
)
SELECT
    o.O_ORDERKEY,
    o.O_ORDERDATE,
    o.O_CUSTKEY,
    o.O_ORDERSTATUS
```

Además, creo varias dimensiones usando joins. Por ejemplo, la dimensión customers:



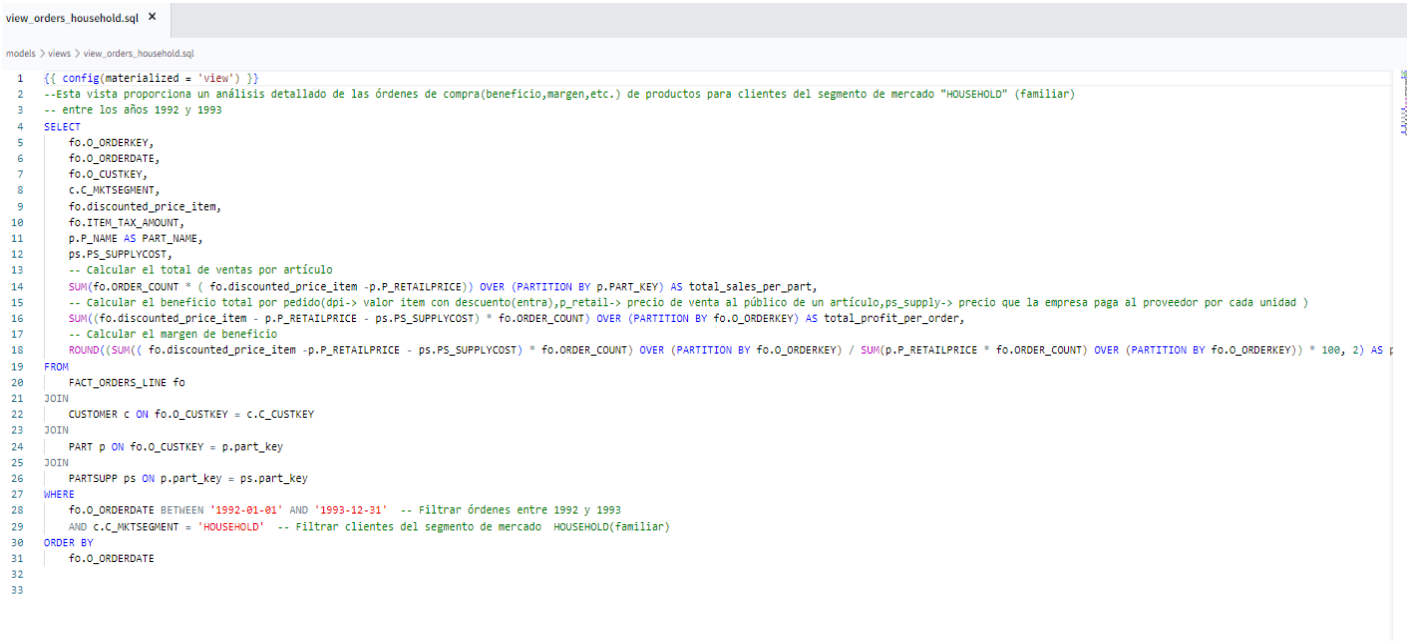
La dimensión suppliers,que agrupa varias tablas:



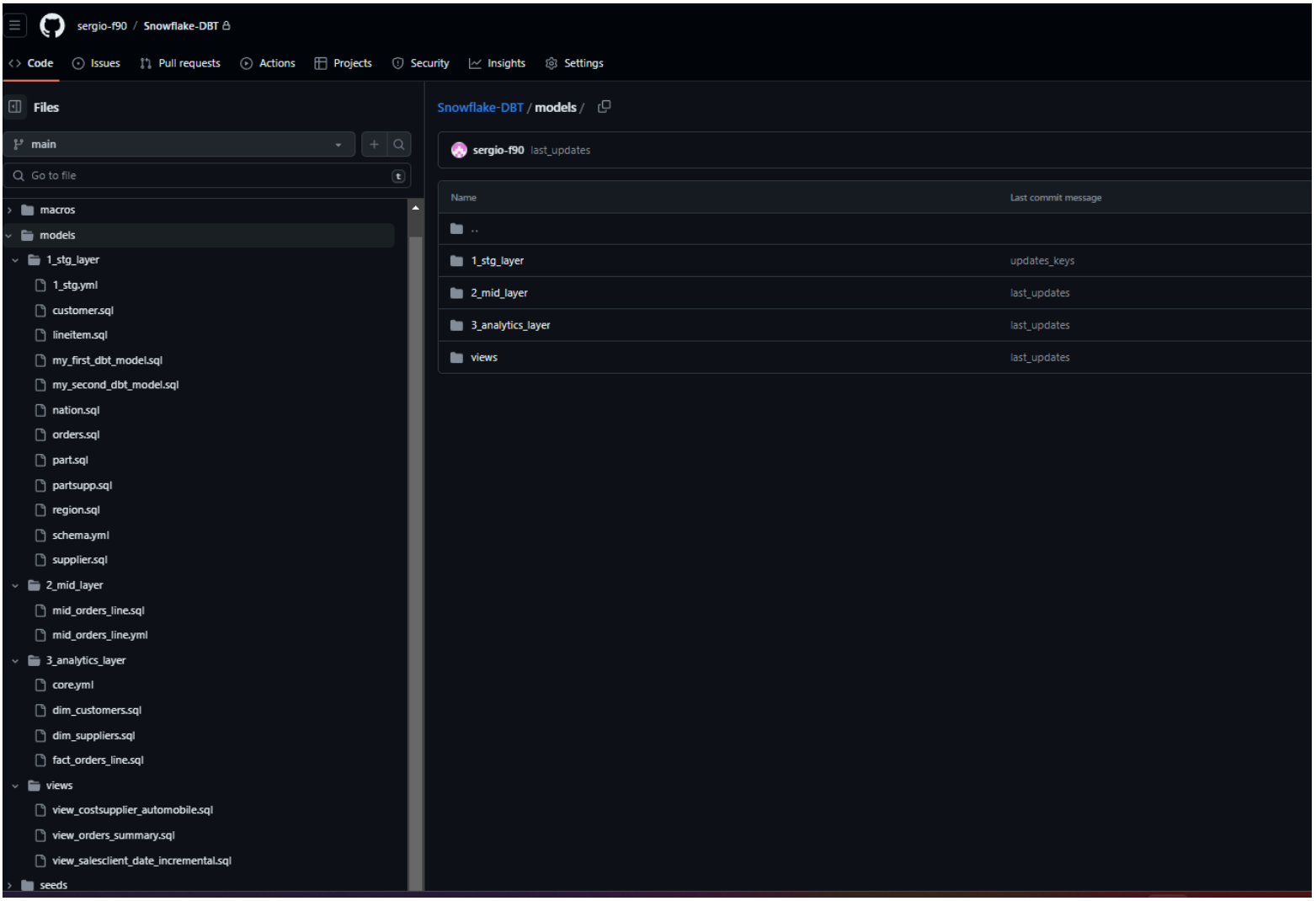
Vistas:

Creación de varias vistas para un análisis en concreto:

-(EJEMPLO) Análisis detallado de las órdenes de compra(beneficio,margen,etc.) de productos para clientes del segmento de mercado "HOUSEHOLD" (familiar) entre 1992 y 1993



Git-Hub


sergio-f90 / Snowflake-DBT
Code
Issues
Pull requests
Actions
Projects
Security
Insights
Settings
Files
main
Go to file
macros
models
1_stg_layer
1_stg.yml
customer.sql
lineitem.sql
my_first_dbt_model.sql
my_second_dbt_model.sql
nation.sql
orders.sql
part.sql
partsupp.sql
region.sql
schema.yml
supplier.sql
2_mid_layer
mid_orders_line.sql
mid_orders_line.yml
3_analytics_layer
core.yml
dim_customers.sql
dim_suppliers.sql
fact_orders_line.sql
views
view_costsupplier_automobile.sql
view_orders_summary.sql
view_salesclient_date_incremental.sql
seeds
Snowflake-DBT / models /
sergio-f90 last_updates
Name
Last commit message
..
1_stg_layer
updates_keys
2_mid_layer
last_updates
3_analytics_layer
last_updates
views
last_updates