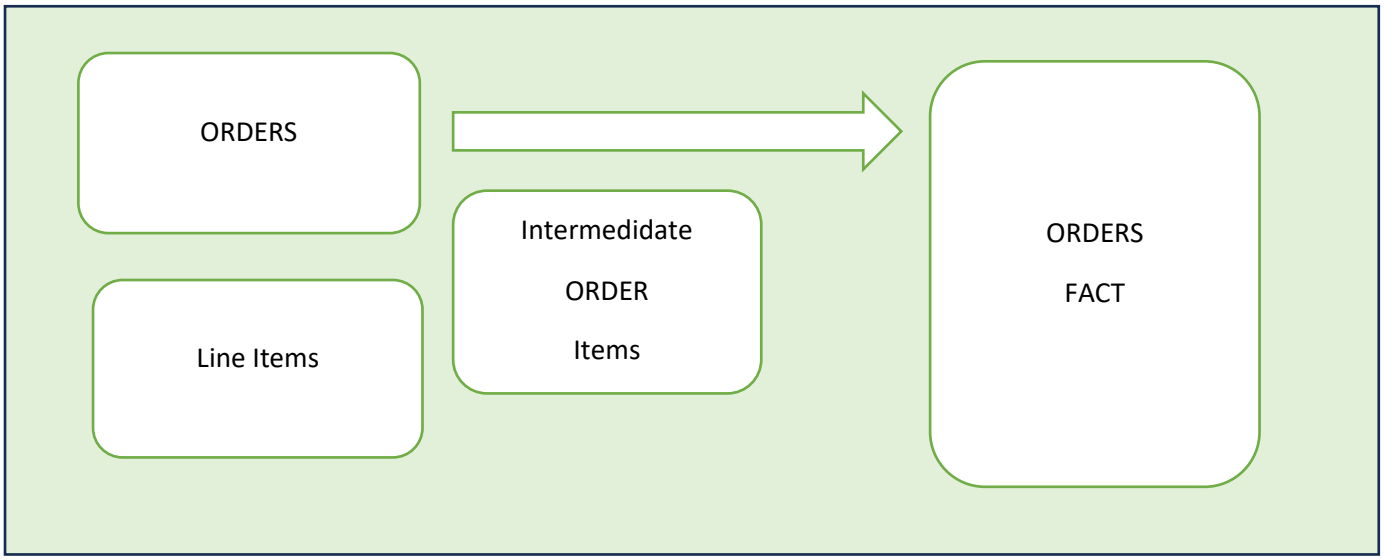
Handson Lab Day 3:

1. Build final MART Models- 1. Orders 2. Line
2. Test the Output in Snowflake for Models- 1. Orders 2. Line Items
3. The **ref** function

Use Case Overview:

- Raw Data: Source Data Set
- Transformations:
- Enriched
- TEST
- Documentation



CREATE Final MART Model:

Step1: intermediate model that performs new line items calculations

`Creating  new file int_order_items.sql`

`models/marts/core/int_order_items.sql`

`Aggregation-  Logic Code`

```
with orders as (

    select * from {{ ref('stg_tpch_orders') }}

),

line_item as (

    select * from {{ ref('stg_tpch_line_items') }}

)
select

    line_item.order_item_key,
    orders.order_key,
    orders.customer_key,
    orders.order_date,
    orders.status_code as order_status_code,
    line_item.part_key,
    line_item.supplier_key,
    line_item.return_flag,
    line_item.line_number,
    line_item.status_code as order_item_status_code,
    line_item.ship_date,
    line_item.commit_date,
    line_item.receipt_date,
    line_item.ship_mode,
    line_item.extended_price,
    line_item.quantity,

    -- extended_price is actually the line item total,
    -- so we back out the extended price per item
    (line_item.extended_price/nullif(line_item.quantity, 0))::decimal(16,2) as
base_price,
    line_item.discount_percentage,
    (base_price * (1 - line_item.discount_percentage))::decimal(16,2) as
discounted_price,

    line_item.extended_price as gross_item_sales_amount,
    (line_item.extended_price * (1 -
line_item.discount_percentage))::decimal(16,2) as
discounted_item_sales_amount,
    -- We model discounts as negative amounts
    (-1 * line_item.extended_price *
line_item.discount_percentage)::decimal(16,2) as item_discount_amount,
    line_item.tax_rate,
    ((gross_item_sales_amount + item_discount_amount) *
line_item.tax_rate)::decimal(16,2) as item_tax_amount,
```

```
    (
        gross_item_sales_amount +
        item_discount_amount +
        item_tax_amount
    )::decimal(16,2) as net_item_sales_amount

from
    orders
inner join line_item
        on orders.order_key = line_item.order_key
order by
    orders.order_date
```

Step2:create Final Fact model.

Start by creating a new file called `fct_orders.sql` with the following file path: `models/marts/core/fct_orders.sql`

Aggregation- Logic Code

```
with orders as (

    select * from {{ ref('stg_tpch_orders') }}

),
order_item as (

    select * from {{ ref('int_order_items') }}

),
order_item_summary as (

    select
        order_key,
        sum(gross_item_sales_amount) as gross_item_sales_amount,
        sum(item_discount_amount) as item_discount_amount,
        sum(item_tax_amount) as item_tax_amount,
        sum(net_item_sales_amount) as net_item_sales_amount
    from order_item
    group by
        1
),
```

```
final as (

    select

        orders.order_key,
        orders.order_date,
        orders.customer_key,
        orders.status_code,
        orders.priority_code,
        orders.clerk_name,
        orders.ship_priority,

        1 as order_count,
        order_item_summary.gross_item_sales_amount,
        order_item_summary.item_discount_amount,
        order_item_summary.item_tax_amount,
        order_item_summary.net_item_sales_amount
    from
        orders
        inner join order_item_summary
            on orders.order_key = order_item_summary.order_key
)
select
    *
from
    final

order by
    order_date
```

Run command: `dbt run --select int_order_items+`

OR

Run command: `dbt run`

Lab3- Final Output:

validate Aggregrations & Tables in Snowflake

PC_DBT_DB / DBT_RN / INT_ORDER_ITEMS

Load Data

Table | PC_DBT_ROLE | 5 minutes ago | 6.0M | 297.3MB

Table Details | Columns | Data Preview | Copy History

COMPUTE_WH | 100 of 6.0M Rows • Updated 1 minute ago

	RICE	DISCOUNT_PERCENTAGE	DISCOUNTED_PRICE	GROSS_ITEM_SALES_AMOUNT	DISCOUNTED_ITEM_SALE
1	8.32	0.02	1331.15	58407.76	
2	7.56	0.10	1473.80	63864.84	
3	6.74	0.10	1563.07	74679.82	
4	8.28	0.10	1186.45	47458.08	
5	4.63	0.05	1543.40	27618.71	
6	7.72	0.09	1563.13	54967.04	
7	1.35	0.03	1242.91	32033.75	
8	4.01	0.02	1052.53	51552.48	
9	5.14	0.04	1156.93	19282.24	
10	8.56	0.08	1351.08	26434.08	

PC_DBT_DB / DBT_RN / FCT_ORDERS

Load Data

Table | PC_DBT_ROLE | 31 minutes ago | 1.5M | 34.3MB

Table Details | Columns | Data Preview | Copy History

COMPUTE_WH | 100 of 1.5M Rows • Updated just now

	COUNT	GROSS_ITEM_SALES_AMOUNT	ITEM_DISCOUNT_AMOUNT	ITEM_TAX_AMOUNT	NET_ITEM_SALES_AM
1	1	145903.30	-3950.71	5381.87	14
2	1	257728.70	-7025.05	11927.48	26
3	1	283622.11	-9432.17	9972.15	28
4	1	6630.64	-265.23	254.62	
5	1	63289.47	-70.35	3862.80	6
6	1	150981.80	-8398.53	3597.94	14
7	1	173097.09	-5080.86	8478.46	17
8	1	135479.70	-5388.31	8440.74	13
9	1	158589.25	-9547.83	3647.14	15
10	1	224816.86	-11815.26	6856.24	21