INSTACART Dimensional modeling in SnowFlake

Raw dataset from Kaggle. there are 5 tables

Orders Table:

Column Name	Data Type	Description
order_id	integer	Unique identifier for an order
user_id	integer	Unique identifier for a user
order_number	integer	A counter for the orders of a user
order_dow	integer	The day of the week the order was placed
order_hour_of_day	integer	The hour of the day the order was placed
days_since_prior_order	integer	Number of days since the previous order

Products Table:

Column Name	Data Type	Description
product_id	integer	Unique identifier for a product
product_name	varchar	Name of the product
aisle_id	integer	Unique identifier for an aisle
department_id	integer	Unique identifier for a department

Order Products Table:

Column Name	Data Type	Description
order_id	integer	Unique identifier for an order
product_id	integer	Unique identifier for a product
add_to_cart_order	integer	The order in which the product was added to the cart
reordered	boolean	Has the product been ordered by this user in the past?

Aisles Table:

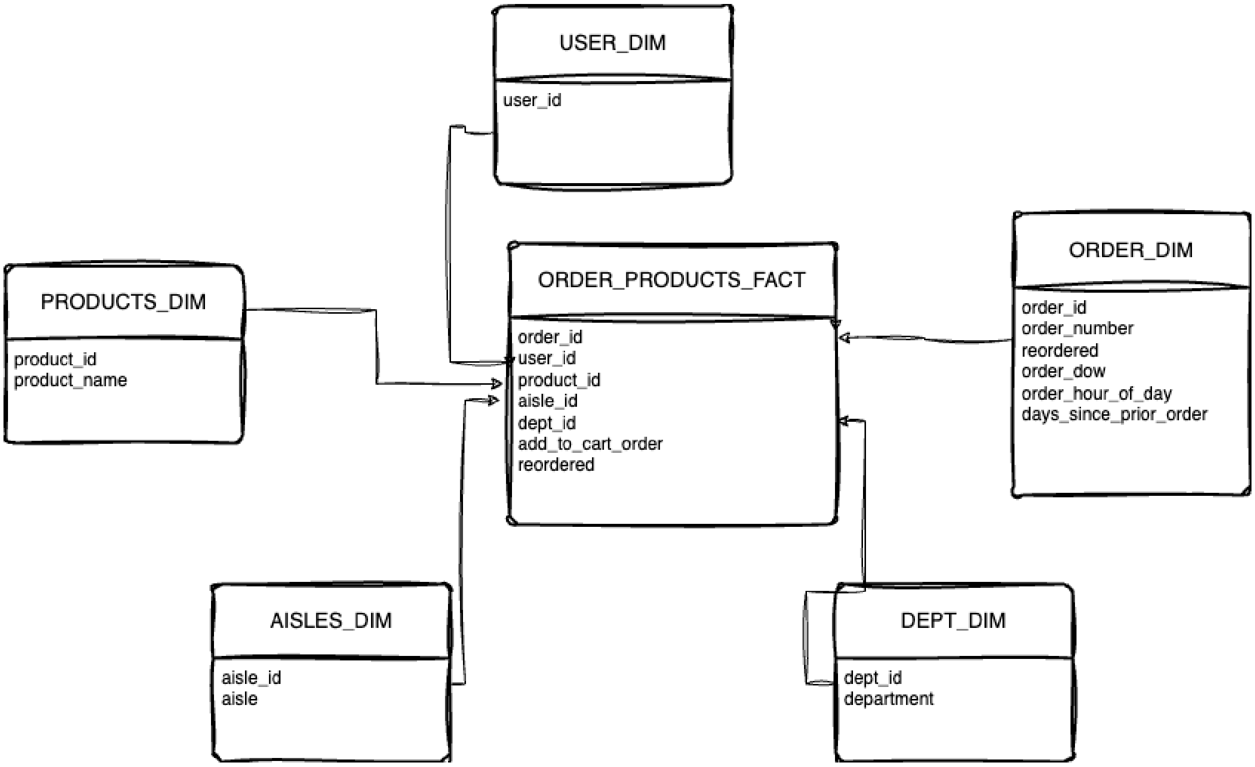
Column Name	Data Type	Description
aisle_id	integer	Unique identifier for an aisle
aisle	varchar	Name of the aisle

Departments Table:

Column Name	Data Type	Description
department_id	integer	Unique identifier for a department
department	varchar	Name of the department

Building a dimensional model from the original data model

Designing a star schema :



Executing same in snowflake

```
-- Establising connection from AWS TO SNOWFLAKE
create stage my_stage
```

```
URL="s3://etldimensionalproject/instacart/"
CREDENTIALS = (AWS_KEY_ID='keyid' AWS_SECRET_KEY='secretkeyid')

-- MENTIONING THE FILE FORMAT
CREATE OR REPLACE FILE FORMAT csv_file_format
TYPE = 'CSV'
FIELD_DELIMITER = ','
SKIP_HEADER = 1
FIELD_OPTIONALLY_ENCLOSED_BY='"';

-- TABLE CREATION OF ALL 5 TABLES
CREATE TABLE aisles (
        aisle_id INTEGER PRIMARY KEY,
        aisle VARCHAR
    );

COPY INTO aisles (aisle_id, aisle)
FROM @my_stage/aisles.csv
FILE_FORMAT = (FORMAT_NAME = 'csv_file_format');


CREATE TABLE departments (
        department_id INTEGER PRIMARY KEY,
        department VARCHAR
    );

COPY INTO departments (department_id, department)
FROM @my_stage/departments.csv
FILE_FORMAT = (FORMAT_NAME = 'csv_file_format');

CREATE OR REPLACE TABLE products (
        product_id INTEGER PRIMARY KEY,
        product_name VARCHAR,
        aisle_id INTEGER,
        department_id INTEGER
    );
```

```
COPY INTO products (product_id, product_name, aisle_id, departme
FROM @my_stage/products.csv
FILE_FORMAT = (FORMAT_NAME = 'csv_file_format');

CREATE OR REPLACE TABLE orders (
        order_id INTEGER PRIMARY KEY,
        user_id INTEGER,
        eval_set STRING,
        order_number INTEGER,
        order_dow INTEGER,
        order_hour_of_day INTEGER,
        days_since_prior_order INTEGER
    );

COPY INTO orders (order_id, user_id, eval_set, order_number, or
FROM @my_stage/orders.csv
FILE_FORMAT = (FORMAT_NAME = 'csv_file_format');


CREATE OR REPLACE TABLE order_products (
        order_id INTEGER,
        product_id INTEGER,
        add_to_cart_order INTEGER,
        reordered INTEGER,
        PRIMARY KEY (order_id, product_id)
    );

COPY INTO order_products (order_id, product_id, add_to_cart_orde
FROM @my_stage/order_products.csv
FILE_FORMAT = (FORMAT_NAME = 'csv_file_format');

--- CREATION OF DIMENSION AND FACT TABLES FROM THE DATA MODEL
CREATE OR REPLACE TABLE dim_users AS (
  SELECT
    user_id
  FROM
    orders
);

CREATE OR REPLACE TABLE aisles_dim AS (
       SELECT aisle_id,aisle from aisles
    );

CREATE OR REPLACE TABLE products_dim AS(
       SELECT product_id,product_name from products
    );

CREATE OR REPLACE TABLE dept_dim AS(
       SELECT department_id,department from departments
    );

CREATE OR REPLACE TABLE orders_dim AS(
       SELECT order_id,
order_number,
order_dow,
order_hour_of_day,
days_since_prior_order from orders
    );

CREATE OR REPLACE TABLE order_products_fact AS(
       SELECT op.order_id,op.product_id,o.user_id,p.department_i
       from order_products op
       join orders o on op.order_id=o.order_id
       join products p on op.product_id=p.product_id
    );

-- ANALYTICS on INSTACART
-- 1. Calculate total no of products per depatment

SELECT department,COUNT(product_id) as total_prod
FROM order_products_fact opf
```

```
join departments d
ON opf.department_id=d.department_id
GROUP BY department;

--Find the top 5 aisles with the highest number of reordered pr
SELECT aisle,sum(case when reordered=1 then 1 end) as reordered_
FROM order_products_fact opf
join aisles a
ON opf.aisle_id=a.aisle_id
GROUP BY  aisle
ORDER BY reordered_prods desc
LIMIT 5;

---- Calculate the average number of products added to the cart
SELECT order_dow,avg(add_to_cart_order)
FROM order_products_fact opf
JOIN orders o
ON opf.order_id=o.order_id
GROUP BY order_dow
ORDER BY order_dow;

-- Calculate the top 10 users with the highest number of unique

with cte as (SELECT user_id,count(distinct product_id) AS unique
FROM order_products_fact opf
GROUP BY user_id
ORDER BY unique_prod DESC)
select user_id, dense_rank() over(order by unique_prod desc) as
FROM cte
qualify rnk <=10;
```