use role sysadmin;

create warehouse snowpark\_etl\_wh

with

warehouse\_size = 'small'

warehouse\_type = 'standard'

auto\_suspend = 60

auto\_resume = true

min\_cluster\_count = 1

max\_cluster\_count = 1 ;

--scaling\_policy = 'standard';

-- create database

create database if not exists sales\_dwh;

use database sales\_dwh;

create schema if not exists source; -- will have source stage etc

create schema if not exists curated; -- data curation and de-duplication

create schema if not exists consumption; -- fact & dimension

create schema if not exists audit; -- to capture all audit records

create schema if not exists common; -- for file formats sequence object etc

use schema source;

create or replace stage my\_internal\_stg;

ls @my\_internal\_stg;

use schema common;

-- create file formats csv (India), json (France), Parquet (USA)

create or replace file format my\_csv\_format

type = csv

field\_delimiter = ','

skip\_header = 1

null\_if = ('null', 'null')

empty\_field\_as\_null = true

field\_optionally\_enclosed\_by = '\042'

compression = auto;

-- json file format with strip outer array true

create or replace file format my\_json\_format

type = json

strip\_outer\_array = true

compression = auto;

-- parquet file format

create or replace file format my\_parquet\_format

type = parquet

compression = snappy;

use schema source;

-- Internal Stage - Query The CSV Data File Format

select

t.$1::text as order\_id,

t.$2::text as customer\_name,

t.$3::text as mobile\_key,

t.$4::number as order\_quantity,

t.$5::number as unit\_price,

t.$6::number as order\_valaue,

t.$7::text as promotion\_code ,

t.$8::number(10,2) as final\_order\_amount,

t.$9::number(10,2) as tax\_amount,

t.$10::date as order\_dt,

t.$11::text as payment\_status,

t.$12::text as shipping\_status,

t.$13::text as payment\_method,

t.$14::text as payment\_provider,

t.$15::text as mobile,

t.$16::text as shipping\_address

from

@sales\_dwh.source.my\_internal\_stg/sales/source=IN/format=csv/

(file\_format => 'sales\_dwh.common.my\_csv\_format') t;

-- Internal Stage - Query The Parquet Data File Format

select

$1:"Order ID"::text as orde\_id,

$1:"Customer Name"::text as customer\_name,

$1:"Mobile Model"::text as mobile\_key,

to\_number($1:"Quantity") as quantity,

to\_number($1:"Price per Unit") as unit\_price,

to\_decimal($1:"Total Price") as total\_price,

$1:"Promotion Code"::text as promotion\_code,

$1:"Order Amount"::number(10,2) as order\_amount,

to\_decimal($1:"Tax") as tax,

$1:"Order Date"::date as order\_dt,

$1:"Payment Status"::text as payment\_status,

$1:"Shipping Status"::text as shipping\_status,

$1:"Payment Method"::text as payment\_method,

$1:"Payment Provider"::text as payment\_provider,

$1:"Phone"::text as phone,

$1:"Delivery Address"::text as shipping\_address

from

@sales\_dwh.source.my\_internal\_stg/sales/source=US/format=parquet/

(file\_format => 'sales\_dwh.common.my\_parquet\_format');

-- Internal Stage - Query The JSON Data File Format

select

$1:"Order ID"::text as orde\_id,

$1:"Customer Name"::text as customer\_name,

$1:"Mobile Model"::text as mobile\_key,

to\_number($1:"Quantity") as quantity,

to\_number($1:"Price per Unit") as unit\_price,

to\_decimal($1:"Total Price") as total\_price,

$1:"Promotion Code"::text as promotion\_code,

$1:"Order Amount"::number(10,2) as order\_amount,

to\_decimal($1:"Tax") as tax,

$1:"Order Date"::date as order\_dt,

$1:"Payment Status"::text as payment\_status,

$1:"Shipping Status"::text as shipping\_status,

$1:"Payment Method"::text as payment\_method,

$1:"Payment Provider"::text as payment\_provider,

$1:"Phone"::text as phone,

$1:"Delivery Address"::text as shipping\_address

from

@sales\_dwh.source.my\_internal\_stg/sales/source=FR/format=json/

(file\_format => sales\_dwh.common.my\_json\_format);

--Step 6- Create Foreign Exchange Rate Data fro converting all sales to USD

use schema source;

ls @my\_internal\_stg;

use schema common;

create or replace transient table exchange\_rate(

date date,

usd2usd decimal(10,7),

usd2eu decimal(10,7),

usd2can decimal(10,7),

usd2uk decimal(10,7),

usd2inr decimal(10,7),

usd2jp decimal(10,7)

);

copy into sales\_dwh.common.exchange\_rate

from

(

select

t.$1::date as exchange\_dt,

to\_decimal(t.$2) as usd2usd,

to\_decimal(t.$3,12,10) as usd2eu,

to\_decimal(t.$4,12,10) as usd2can,

to\_decimal(t.$4,12,10) as usd2uk,

to\_decimal(t.$4,12,10) as usd2inr,

to\_decimal(t.$4,12,10) as usd2jp

from

@sales\_dwh.source.my\_internal\_stg/exchange/exchange-rate-data.csv

(file\_format => 'sales\_dwh.common.my\_csv\_format') t

);

--Step-7 Loading Data From Internal Stage to Source Tables

use schema source;

--Create sequences for snowflake primary key identifier for each region

create or replace sequence in\_sales\_order\_seq

start = 1

increment = 1

comment='This is sequence for India sales order table';

create or replace sequence us\_sales\_order\_seq

start = 1

increment = 1

comment='This is sequence for USA sales order table';

create or replace sequence fr\_sales\_order\_seq

start = 1

increment = 1

comment='This is sequence for France sales order table';

show sequences;

-- India Sales Table in Source Schema (CSV File)

create or replace transient table in\_sales\_order (

sales\_order\_key number(38,0),

order\_id varchar(),

customer\_name varchar(),

mobile\_key varchar(),

order\_quantity number(38,0),

unit\_price number(38,0),

order\_valaue number(38,0),

promotion\_code varchar(),

final\_order\_amount number(10,2),

tax\_amount number(10,2),

order\_dt date,

payment\_status varchar(),

shipping\_status varchar(),

payment\_method varchar(),

payment\_provider varchar(),

mobile varchar(),

shipping\_address varchar(),

\_metadata\_file\_name varchar(),

\_metadata\_row\_numer number(38,0),

\_metadata\_last\_modified timestamp\_ntz(9)

);

-- US Sales Table in Source Schema (Parquet File)

create or replace transient table us\_sales\_order (

sales\_order\_key number(38,0),

order\_id varchar(),

customer\_name varchar(),

mobile\_key varchar(),

order\_quantity number(38,0),

unit\_price number(38,0),

order\_valaue number(38,0),

promotion\_code varchar(),

final\_order\_amount number(10,2),

tax\_amount number(10,2),

order\_dt date,

payment\_status varchar(),

shipping\_status varchar(),

payment\_method varchar(),

payment\_provider varchar(),

phone varchar(),

shipping\_address varchar(),

\_metadata\_file\_name varchar(),

\_metadata\_row\_numer number(38,0),

\_metadata\_last\_modified timestamp\_ntz(9)

);

-- France Sales Table in Source Schema (JSON File)

create or replace transient table fr\_sales\_order (

sales\_order\_key number(38,0),

order\_id varchar(),

customer\_name varchar(),

mobile\_key varchar(),

order\_quantity number(38,0),

unit\_price number(38,0),

order\_valaue number(38,0),

promotion\_code varchar(),

final\_order\_amount number(10,2),

tax\_amount number(10,2),

order\_dt date,

payment\_status varchar(),

shipping\_status varchar(),

payment\_method varchar(),

payment\_provider varchar(),

phone varchar(),

shipping\_address varchar(),

\_metadata\_file\_name varchar(),

\_metadata\_row\_numer number(38,0),

\_metadata\_last\_modified timestamp\_ntz(9)

);

-- Copy data into source snowflake table

copy into sales\_dwh.source.in\_sales\_order from

(select

in\_sales\_order\_seq.nextval,

t.$1::text as order\_id,

t.$2::text as customer\_name,

t.$3::text as mobile\_key,

t.$4::number as order\_quantity,

t.$5::number as unit\_price,

t.$6::number as order\_valaue,

t.$7::text as promotion\_code ,

t.$8::number(10,2) as final\_order\_amount,

t.$9::number(10,2) as tax\_amount,

t.$10::date as order\_dt,

t.$11::text as payment\_status,

t.$12::text as shipping\_status,

t.$13::text as payment\_method,

t.$14::text as payment\_provider,

t.$15::text as mobile,

t.$16::text as shipping\_address,

metadata$filename as stg\_file\_name,

metadata$file\_row\_number as stg\_row\_numer,

metadata$file\_last\_modified as stg\_last\_modified

from

@sales\_dwh.source.my\_internal\_stg/sales/source=IN/format=csv

(file\_format => 'sales\_dwh.common.my\_csv\_format') t

);

copy into sales\_dwh.source.us\_sales\_order

from

(

select

us\_sales\_order\_seq.nextval,

$1:"Order ID"::text as orde\_id,

$1:"Customer Name"::text as customer\_name,

$1:"Mobile Model"::text as mobile\_key,

to\_number($1:"Quantity") as quantity,

to\_number($1:"Price per Unit") as unit\_price,

to\_decimal($1:"Total Price") as total\_price,

$1:"Promotion Code"::text as promotion\_code,

$1:"Order Amount"::number(10,2) as order\_amount,

to\_decimal($1:"Tax") as tax,

$1:"Order Date"::date as order\_dt,

$1:"Payment Status"::text as payment\_status,

$1:"Shipping Status"::text as shipping\_status,

$1:"Payment Method"::text as payment\_method,

$1:"Payment Provider"::text as payment\_provider,

$1:"Phone"::text as phone,

$1:"Delivery Address"::text as shipping\_address,

metadata$filename as stg\_file\_name,

metadata$file\_row\_number as stg\_row\_numer,

metadata$file\_last\_modified as stg\_last\_modified

from

@sales\_dwh.source.my\_internal\_stg/sales/source=US/format=parquet

(file\_format => sales\_dwh.common.my\_parquet\_format)

);

copy into sales\_dwh.source.fr\_sales\_order

from

(

select

sales\_dwh.source.fr\_sales\_order\_seq.nextval,

$1:"Order ID"::text as orde\_id,

$1:"Customer Name"::text as customer\_name,

$1:"Mobile Model"::text as mobile\_key,

to\_number($1:"Quantity") as quantity,

to\_number($1:"Price per Unit") as unit\_price,

to\_decimal($1:"Total Price") as total\_price,

$1:"Promotion Code"::text as promotion\_code,

$1:"Order Amount"::number(10,2) as order\_amount,

to\_decimal($1:"Tax") as tax,

$1:"Order Date"::date as order\_dt,

$1:"Payment Status"::text as payment\_status,

$1:"Shipping Status"::text as shipping\_status,

$1:"Payment Method"::text as payment\_method,

$1:"Payment Provider"::text as payment\_provider,

$1:"Phone"::text as phone,

$1:"Delivery Address"::text as shipping\_address ,

metadata$filename as stg\_file\_name,

metadata$file\_row\_number as stg\_row\_numer,

metadata$file\_last\_modified as stg\_last\_modified

from

@sales\_dwh.source.my\_internal\_stg/sales/source=FR/format=json

(file\_format => sales\_dwh.common.my\_json\_format)

);

select \* from fr\_sales\_order;

select \* from in\_sales\_order;

select \* from us\_sales\_order;

--Step-8 Loading Data From Source To Curated Layer (Bronze to silver layer)

--Create sequence objects for snwflake keys

use schema curated;

create or replace sequence in\_sales\_order\_seq

start = 1

increment = 1

comment='This is sequence for India sales order table';

create or replace sequence us\_sales\_order\_seq

start = 1

increment = 1

comment='This is sequence for USA sales order table';

create or replace sequence fr\_sales\_order\_seq

start = 1

increment = 1

comment='This is sequence for France sales order table';

--Create Silver tables

use schema curated;

-- curated India sales order table

create or replace table in\_sales\_order (

sales\_order\_key number(38,0),

order\_id varchar(),

order\_dt date,

customer\_name varchar(),

mobile\_key varchar(),

country varchar(),

region varchar(),

order\_quantity number(38,0),

local\_currency varchar(),

local\_unit\_price number(38,0),

promotion\_code varchar(),

local\_total\_order\_amt number(10,2),

local\_tax\_amt number(10,2),

exhchange\_rate number(15,7),

us\_total\_order\_amt number(23,8),

usd\_tax\_amt number(23,8),

payment\_status varchar(),

shipping\_status varchar(),

payment\_method varchar(),

payment\_provider varchar(),

conctact\_no varchar(),

shipping\_address varchar()

);

-- curated US sales order table

create or replace table us\_sales\_order (

sales\_order\_key number(38,0),

order\_id varchar(),

order\_dt date,

customer\_name varchar(),

mobile\_key varchar(),

country varchar(),

region varchar(),

order\_quantity number(38,0),

local\_currency varchar(),

local\_unit\_price number(38,0),

promotion\_code varchar(),

local\_total\_order\_amt number(10,2),

local\_tax\_amt number(10,2),

exhchange\_rate number(15,7),

us\_total\_order\_amt number(23,8),

usd\_tax\_amt number(23,8),

payment\_status varchar(),

shipping\_status varchar(),

payment\_method varchar(),

payment\_provider varchar(),

conctact\_no varchar(),

shipping\_address varchar()

);

-- -- curated FR sales order table

create or replace table fr\_sales\_order (

sales\_order\_key number(38,0),

order\_id varchar(),

order\_dt date,

customer\_name varchar(),

mobile\_key varchar(),

country varchar(),

region varchar(),

order\_quantity number(38,0),

local\_currency varchar(),

local\_unit\_price number(38,0),

promotion\_code varchar(),

local\_total\_order\_amt number(10,2),

local\_tax\_amt number(10,2),

exhchange\_rate number(15,7),

us\_total\_order\_amt number(23,8),

usd\_tax\_amt number(23,8),

payment\_status varchar(),

shipping\_status varchar(),

payment\_method varchar(),

payment\_provider varchar(),

conctact\_no varchar(),

shipping\_address varchar()

);

-- region dimension

use schema consumption;

create or replace sequence region\_dim\_seq start = 1 increment = 1;

create or replace transient table region\_dim(

region\_id\_pk number primary key,

Country text,

Region text,

isActive text(1)

);

create or replace sequence product\_dim\_seq start = 1 increment = 1;

create or replace transient table product\_dim(

product\_id\_pk number primary key,

Mobile\_key text,

Brand text,

Model text,

Color text,

Memory text,

isActive text(1)

);

create or replace sequence promo\_code\_dim\_seq start = 1 increment = 1;

create or replace transient table promo\_code\_dim(

promo\_code\_id\_pk number primary key,

promo\_code text,

isActive text(1)

);

create or replace sequence customer\_dim\_seq start = 1 increment = 1;

create or replace transient table customer\_dim(

customer\_id\_pk number primary key,

customer\_name text,

CONCTACT\_NO text,

SHIPPING\_ADDRESS text,

country text,

region text,

isActive text(1)

);

create or replace sequence payment\_dim\_seq start = 1 increment = 1;

create or replace transient table payment\_dim(

payment\_id\_pk number primary key,

PAYMENT\_METHOD text,

PAYMENT\_PROVIDER text,

country text,

region text,

isActive text(1)

);

create or replace sequence date\_dim\_seq start = 1 increment = 1;

create or replace transient table date\_dim(

date\_id\_pk int primary key,

order\_dt date,

order\_year int,

oder\_month int,

order\_quater int,

order\_day int,

order\_dayofweek int,

order\_dayname text,

order\_dayofmonth int,

order\_weekday text

);

create or replace table sales\_fact (

order\_id\_pk number(38,0),

order\_code varchar(),

date\_id\_fk number(38,0),

region\_id\_fk number(38,0),

customer\_id\_fk number(38,0),

payment\_id\_fk number(38,0),

product\_id\_fk number(38,0),

promo\_code\_id\_fk number(38,0),

order\_quantity number(38,0),

local\_total\_order\_amt number(10,2),

local\_tax\_amt number(10,2),

exhchange\_rate number(15,7),

us\_total\_order\_amt number(23,8),

usd\_tax\_amt number(23,8)

);

alter table sales\_fact add

constraint fk\_sales\_region FOREIGN KEY (REGION\_ID\_FK) REFERENCES region\_dim (REGION\_ID\_PK) NOT ENFORCED;

alter table sales\_fact add

constraint fk\_sales\_date FOREIGN KEY (DATE\_ID\_FK) REFERENCES date\_dim (DATE\_ID\_PK) NOT ENFORCED;

alter table sales\_fact add

constraint fk\_sales\_customer FOREIGN KEY (CUSTOMER\_ID\_FK) REFERENCES customer\_dim (CUSTOMER\_ID\_PK) NOT ENFORCED;

--

alter table sales\_fact add

constraint fk\_sales\_payment FOREIGN KEY (PAYMENT\_ID\_FK) REFERENCES payment\_dim (PAYMENT\_ID\_PK) NOT ENFORCED;

alter table sales\_fact add

constraint fk\_sales\_product FOREIGN KEY (PRODUCT\_ID\_FK) REFERENCES product\_dim (PRODUCT\_ID\_PK) NOT ENFORCED;

alter table sales\_fact add

constraint fk\_sales\_promot FOREIGN KEY (PROMO\_CODE\_ID\_FK) REFERENCES promo\_code\_dim (PROMO\_CODE\_ID\_PK) NOT ENFORCED;