This is not mandatory reading, but here's the code we'll run in the "Stages and Basic Ingestion" videos. It may come in handy when you're doing the associated hands-on assignment.

USE ROLE accountadmin;

---> create tasty\_bytes database

CREATE OR REPLACE DATABASE tasty\_bytes;

---> create raw\_pos schema

CREATE OR REPLACE SCHEMA tasty\_bytes.raw\_pos;

---> create raw\_customer schema

CREATE OR REPLACE SCHEMA tasty\_bytes.raw\_customer;

---> create harmonized schema

CREATE OR REPLACE SCHEMA tasty\_bytes.harmonized;

---> create analytics schema

CREATE OR REPLACE SCHEMA tasty\_bytes.analytics;

---> create warehouses

CREATE OR REPLACE WAREHOUSE demo\_build\_wh

WAREHOUSE\_SIZE = 'xxxlarge'

WAREHOUSE\_TYPE = 'standard'

AUTO\_SUSPEND = 60

AUTO\_RESUME = TRUE

INITIALLY\_SUSPENDED = TRUE

COMMENT = 'demo build warehouse for tasty bytes assets';

CREATE OR REPLACE WAREHOUSE tasty\_de\_wh

WAREHOUSE\_SIZE = 'xsmall'

WAREHOUSE\_TYPE = 'standard'

AUTO\_SUSPEND = 60

AUTO\_RESUME = TRUE

INITIALLY\_SUSPENDED = TRUE

COMMENT = 'data engineering warehouse for tasty bytes';

USE WAREHOUSE tasty\_de\_wh;

---> file format creation

CREATE OR REPLACE FILE FORMAT tasty\_bytes.public.csv\_ff

type = 'csv';

---> stage creation

CREATE OR REPLACE STAGE tasty\_bytes.public.s3load

url = 's3://sfquickstarts/frostbyte\_tastybytes/'

file\_format = tasty\_bytes.public.csv\_ff;

---> example of creating an internal stage

-- CREATE OR REPLACE STAGE tasty\_bytes.public.internal\_stage\_test;

---> list files in stage

ls @tasty\_bytes.public.s3load;

---> country table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.country

(

country\_id NUMBER(18,0),

country VARCHAR(16777216),

iso\_currency VARCHAR(3),

iso\_country VARCHAR(2),

city\_id NUMBER(19,0),

city VARCHAR(16777216),

city\_population VARCHAR(16777216)

);

---> franchise table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.franchise

(

franchise\_id NUMBER(38,0),

first\_name VARCHAR(16777216),

last\_name VARCHAR(16777216),

city VARCHAR(16777216),

country VARCHAR(16777216),

e\_mail VARCHAR(16777216),

phone\_number VARCHAR(16777216)

);

---> location table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.location

(

location\_id NUMBER(19,0),

placekey VARCHAR(16777216),

location VARCHAR(16777216),

city VARCHAR(16777216),

region VARCHAR(16777216),

iso\_country\_code VARCHAR(16777216),

country VARCHAR(16777216)

);

---> menu table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.menu

(

menu\_id NUMBER(19,0),

menu\_type\_id NUMBER(38,0),

menu\_type VARCHAR(16777216),

truck\_brand\_name VARCHAR(16777216),

menu\_item\_id NUMBER(38,0),

menu\_item\_name VARCHAR(16777216),

item\_category VARCHAR(16777216),

item\_subcategory VARCHAR(16777216),

cost\_of\_goods\_usd NUMBER(38,4),

sale\_price\_usd NUMBER(38,4),

menu\_item\_health\_metrics\_obj VARIANT

);

---> truck table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.truck

(

truck\_id NUMBER(38,0),

menu\_type\_id NUMBER(38,0),

primary\_city VARCHAR(16777216),

region VARCHAR(16777216),

iso\_region VARCHAR(16777216),

country VARCHAR(16777216),

iso\_country\_code VARCHAR(16777216),

franchise\_flag NUMBER(38,0),

year NUMBER(38,0),

make VARCHAR(16777216),

model VARCHAR(16777216),

ev\_flag NUMBER(38,0),

franchise\_id NUMBER(38,0),

truck\_opening\_date DATE

);

---> order\_header table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.order\_header

(

order\_id NUMBER(38,0),

truck\_id NUMBER(38,0),

location\_id FLOAT,

customer\_id NUMBER(38,0),

discount\_id VARCHAR(16777216),

shift\_id NUMBER(38,0),

shift\_start\_time TIME(9),

shift\_end\_time TIME(9),

order\_channel VARCHAR(16777216),

order\_ts TIMESTAMP\_NTZ(9),

served\_ts VARCHAR(16777216),

order\_currency VARCHAR(3),

order\_amount NUMBER(38,4),

order\_tax\_amount VARCHAR(16777216),

order\_discount\_amount VARCHAR(16777216),

order\_total NUMBER(38,4)

);

---> order\_detail table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.order\_detail

(

order\_detail\_id NUMBER(38,0),

order\_id NUMBER(38,0),

menu\_item\_id NUMBER(38,0),

discount\_id VARCHAR(16777216),

line\_number NUMBER(38,0),

quantity NUMBER(5,0),

unit\_price NUMBER(38,4),

price NUMBER(38,4),

order\_item\_discount\_amount VARCHAR(16777216)

);

---> customer loyalty table build

CREATE OR REPLACE TABLE tasty\_bytes.raw\_customer.customer\_loyalty

(

customer\_id NUMBER(38,0),

first\_name VARCHAR(16777216),

last\_name VARCHAR(16777216),

city VARCHAR(16777216),

country VARCHAR(16777216),

postal\_code VARCHAR(16777216),

preferred\_language VARCHAR(16777216),

gender VARCHAR(16777216),

favourite\_brand VARCHAR(16777216),

marital\_status VARCHAR(16777216),

children\_count VARCHAR(16777216),

sign\_up\_date DATE,

birthday\_date DATE,

e\_mail VARCHAR(16777216),

phone\_number VARCHAR(16777216)

);

---> orders\_v view

CREATE OR REPLACE VIEW tasty\_bytes.harmonized.orders\_v

AS

SELECT

oh.order\_id,

oh.truck\_id,

oh.order\_ts,

od.order\_detail\_id,

od.line\_number,

m.truck\_brand\_name,

m.menu\_type,

t.primary\_city,

t.region,

t.country,

t.franchise\_flag,

t.franchise\_id,

f.first\_name AS franchisee\_first\_name,

f.last\_name AS franchisee\_last\_name,

l.location\_id,

cl.customer\_id,

cl.first\_name,

cl.last\_name,

cl.e\_mail,

cl.phone\_number,

cl.children\_count,

cl.gender,

cl.marital\_status,

od.menu\_item\_id,

m.menu\_item\_name,

od.quantity,

od.unit\_price,

od.price,

oh.order\_amount,

oh.order\_tax\_amount,

oh.order\_discount\_amount,

oh.order\_total

FROM tasty\_bytes.raw\_pos.order\_detail od

JOIN tasty\_bytes.raw\_pos.order\_header oh

ON od.order\_id = oh.order\_id

JOIN tasty\_bytes.raw\_pos.truck t

ON oh.truck\_id = t.truck\_id

JOIN tasty\_bytes.raw\_pos.menu m

ON od.menu\_item\_id = m.menu\_item\_id

JOIN tasty\_bytes.raw\_pos.franchise f

ON t.franchise\_id = f.franchise\_id

JOIN tasty\_bytes.raw\_pos.location l

ON oh.location\_id = l.location\_id

LEFT JOIN tasty\_bytes.raw\_customer.customer\_loyalty cl

ON oh.customer\_id = cl.customer\_id;

---> loyalty\_metrics\_v view

CREATE OR REPLACE VIEW tasty\_bytes.harmonized.customer\_loyalty\_metrics\_v

AS

SELECT

cl.customer\_id,

cl.city,

cl.country,

cl.first\_name,

cl.last\_name,

cl.phone\_number,

cl.e\_mail,

SUM(oh.order\_total) AS total\_sales,

ARRAY\_AGG(DISTINCT oh.location\_id) AS visited\_location\_ids\_array

FROM tasty\_bytes.raw\_customer.customer\_loyalty cl

JOIN tasty\_bytes.raw\_pos.order\_header oh

ON cl.customer\_id = oh.customer\_id

GROUP BY cl.customer\_id, cl.city, cl.country, cl.first\_name,

cl.last\_name, cl.phone\_number, cl.e\_mail;

---> orders\_v view

CREATE OR REPLACE VIEW tasty\_bytes.analytics.orders\_v

COMMENT = 'Tasty Bytes Order Detail View'

AS

SELECT DATE(o.order\_ts) AS date, \* FROM tasty\_bytes.harmonized.orders\_v o;

---> customer\_loyalty\_metrics\_v view

CREATE OR REPLACE VIEW tasty\_bytes.analytics.customer\_loyalty\_metrics\_v

COMMENT = 'Tasty Bytes Customer Loyalty Member Metrics View'

AS

SELECT \* FROM tasty\_bytes.harmonized.customer\_loyalty\_metrics\_v;

USE WAREHOUSE demo\_build\_wh;

---> country table load

COPY INTO tasty\_bytes.raw\_pos.country

FROM @tasty\_bytes.public.s3load/raw\_pos/country/;

---> franchise table load

COPY INTO tasty\_bytes.raw\_pos.franchise

FROM @tasty\_bytes.public.s3load/raw\_pos/franchise/;

---> location table load

COPY INTO tasty\_bytes.raw\_pos.location

FROM @tasty\_bytes.public.s3load/raw\_pos/location/;

---> menu table load

COPY INTO tasty\_bytes.raw\_pos.menu

FROM @tasty\_bytes.public.s3load/raw\_pos/menu/;

---> truck table load

COPY INTO tasty\_bytes.raw\_pos.truck

FROM @tasty\_bytes.public.s3load/raw\_pos/truck/;

---> customer\_loyalty table load

COPY INTO tasty\_bytes.raw\_customer.customer\_loyalty

FROM @tasty\_bytes.public.s3load/raw\_customer/customer\_loyalty/;

---> order\_header table load

COPY INTO tasty\_bytes.raw\_pos.order\_header

FROM @tasty\_bytes.public.s3load/raw\_pos/order\_header/;

---> order\_detail table load

COPY INTO tasty\_bytes.raw\_pos.order\_detail

FROM @tasty\_bytes.public.s3load/raw\_pos/order\_detail/;

---> drop demo\_build\_wh

DROP WAREHOUSE IF EXISTS demo\_build\_wh;

USE WAREHOUSE TASTY\_DE\_WH;

SELECT file\_name, error\_count, status, last\_load\_time FROM snowflake.account\_usage.copy\_history

ORDER BY last\_load\_time DESC

LIMIT 10;