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

---> create a clone of the truck table

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

CLONE tasty\_bytes.raw\_pos.truck;

/\* look at metadata for the truck and truck\_clone tables from the table\_storage\_metrics view in the information\_schema \*/

SELECT \* FROM TASTY\_BYTES.INFORMATION\_SCHEMA.TABLE\_STORAGE\_METRICS

WHERE TABLE\_NAME = 'TRUCK\_CLONE' OR TABLE\_NAME = 'TRUCK';

/\* look at metadata for the truck and truck\_clone tables from the tables view in the information\_schema \*/

SELECT \* FROM TASTY\_BYTES.INFORMATION\_SCHEMA.TABLES

WHERE TABLE\_NAME = 'TRUCK\_CLONE' OR TABLE\_NAME = 'TRUCK';

---> insert the truck table into the clone (thus doubling the clone’s size!)

INSERT INTO tasty\_bytes.raw\_pos.truck\_clone

SELECT \* FROM tasty\_bytes.raw\_pos.truck;

---> now use the tables view to look at metadata for the truck and truck\_clone tables again

SELECT \* FROM TASTY\_BYTES.INFORMATION\_SCHEMA.TABLES

WHERE TABLE\_NAME = 'TRUCK\_CLONE' OR TABLE\_NAME = 'TRUCK';

---> clone a schema

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

CLONE tasty\_bytes.raw\_pos;

---> clone a database

CREATE OR REPLACE DATABASE tasty\_bytes\_clone

CLONE tasty\_bytes;

---> clone a table based on an offset (so the table as it was at a certain interval in the past)

CREATE OR REPLACE TABLE tasty\_bytes.raw\_pos.truck\_clone\_time\_travel

CLONE tasty\_bytes.raw\_pos.truck AT(OFFSET => -60\*10);

SELECT \* FROM tasty\_bytes.raw\_pos.truck\_clone\_time\_travel;