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

SHOW TABLES;

---> set the data retention time to 90 days

ALTER TABLE TASTY\_BYTES.RAW\_POS.TEST\_MENU SET DATA\_RETENTION\_TIME\_IN\_DAYS = 90;

SHOW TABLES;

---> set the data retention time to 1 day

ALTER TABLE TASTY\_BYTES.RAW\_POS.TEST\_MENU SET DATA\_RETENTION\_TIME\_IN\_DAYS = 1;

---> clone the truck table

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

CLONE tasty\_bytes.raw\_pos.truck;

SELECT

t.truck\_id,

t.year,

t.make,

t.model

FROM tasty\_bytes.raw\_pos.truck\_dev t;

---> see how the age should have been calculated

SELECT

t.truck\_id,

t.year,

t.make,

t.model,

(YEAR(CURRENT\_DATE()) - t.year) AS truck\_age

FROM tasty\_bytes.raw\_pos.truck\_dev t;

---> record the most recent query\_id, back when the data was still correct

SET good\_data\_query\_id = LAST\_QUERY\_ID();

---> view the variable’s value

SELECT $good\_data\_query\_id;

---> record the time, back when the data was still correct

SET good\_data\_timestamp = CURRENT\_TIMESTAMP;

---> view the variable’s value

SELECT $good\_data\_timestamp;

---> confirm that that worked

SHOW VARIABLES;

---> make the first mistake: calculating the truck’s age incorrectly

SELECT

t.truck\_id,

t.year,

t.make,

t.model,

(YEAR(CURRENT\_DATE()) / t.year) AS truck\_age

FROM tasty\_bytes.raw\_pos.truck\_dev t;

---> make the second mistake: calculate age wrong, and overwrite the year!

UPDATE tasty\_bytes.raw\_pos.truck\_dev t

SET t.year = (YEAR(CURRENT\_DATE()) / t.year);

SELECT

t.truck\_id,

t.year,

t.make,

t.model

FROM tasty\_bytes.raw\_pos.truck\_dev t;

---> select the data as of a particular timestamp

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

AT(TIMESTAMP => $good\_data\_timestamp);

SELECT $good\_data\_timestamp;

---> example code, without a timestamp inserted:

-- SELECT \* FROM tasty\_bytes.raw\_pos.truck\_dev

-- AT(TIMESTAMP => '[insert timestamp]'::TIMESTAMP\_LTZ);

--->example code, with a timestamp inserted

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

AT(TIMESTAMP => '2024-04-04 21:34:31.833 -0700'::TIMESTAMP\_LTZ);

---> calculate the right offset

SELECT TIMESTAMPDIFF(second,CURRENT\_TIMESTAMP,$good\_data\_timestamp);

---> Example code, without an offset inserted:

-- SELECT \* FROM tasty\_bytes.raw\_pos.truck\_dev

-- AT(OFFSET => -[WRITE OFFSET SECONDS PLUS A BIT]);

---> select the data as of a particular number of seconds back in time

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

AT(OFFSET => -45);

SELECT $good\_data\_query\_id;

---> select the data as of its state before a previous query was run

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

BEFORE(STATEMENT => $good\_data\_query\_id);