/\*

Here is a comment explaining the query

\*/

SELECT

store\_status,

COUNT(\*) --how many are there?

FROM stores

GROUP BY store\_status;

SELECT \*

FROM products

WHERE category\_name IN ('TEQUILA', '%SCOTCH%', 'SCOTCH WHISKIES')

LIMIT 100;

SELECT \*

FROM products

WHERE proof::integer != 80

LIMIT 100;

SELECT \*

FROM products

WHERE NOT proof::integer = 80

LIMIT 100;

SELECT \*

FROM producst

WHERE category\_name ILIKE '%whisk%'

LIMIT 100;

SELECT \*

FROM products

WHERE pack BETWEEN 6 AND 12 --inclusive inequality, i.e. <= and >=

LIMIT 100;

SELECT \*

FROM products

WHERE case\_cost > 100

LIMIT 100;

SELECT \*

FROM products

WHERE case\_cost > 100 AND category\_name LIKE '%TEQUILA%'

LIMIT 100;

SELECT \*

FROM products

WHERE case\_cost > 100 AND (category\_name LIKE 'TEQUILA' OR category\_name LIKE 'SCOTCH WHISKIES')

LIMIT 100;

SELECT \*

FROM products

WHERE case\_cost BETWEEN 100 AND 120 AND (category\_name LIKE 'TEQUILA' OR category\_name LIKE 'SCOTCH WHISKIES')

LIMIT 100;

SELECT \*

FROM products

WHERE bottle\_price > 100::money AND category\_name LIKE '%WHISK%'

LIMIT 100;

SELECT \*

FROM products

WHERE bottle\_price::decimal > 100 AND category\_name LIKE '%WHISK%'

LIMIT 100;

SELECT \*

FROM products

WHERE bottle\_price BETWEEN 100::money AND 150::money AND category\_name LIKE '%WHISK%'

LIMIT 100;

SELECT \*

FROM products

WHERE bottle\_price BETWEEN 100::money AND 120::money AND NOT category\_name LIKE '%TEQUILA'

LIMIT 100;

/\* “From the Iowa Liquor Sales Database, I only want information about

Vendor 305. Can you get me the bottle price and proof? The price should

be less than 5 OR the proof should be greater than 100, either is fine.”\*/

SELECT

bottle\_price,

proof

FROM products

WHERE vendor = 305 AND (bottle\_price::decimal < 5 OR proof::integer > 100)

LIMIT 100;

SELECT

store\_status,

COUNT(\*)

FROM stores

WHERE store\_address ILIKE '%des moines%' --filters before aggregating

GROUP BY store\_status

HAVING COUNT(\*) > 100; --filters the aggregated groups

SELECT AVG(btl\_price::numeric)::money

FROM sales

LIMIT 100;

SELECT ROUND(AVG(btl\_price::numeric), 2) AS average\_btl\_price

FROM sales

LIMIT 100;

SELECT COUNT(\*)

FROM sales;

--find stores whose total sales exceed 100000

SELECT

store,

SUM(total)

FROM sales

WHERE county = 'Adair'

GROUP BY store

HAVING SUM(total) > 100000

ORDER BY 2 ASC

LIMIT 100;

SELECT MAX(total) FROM sales;

--find more info on the biggest sale

SELECT \*

FROM sales

ORDER BY total DESC

LIMIT 2;

SELECT ROUND(AVG(state\_btl\_cost::decimal), 2) AS average\_state\_btl\_cost FROM sales;

SELECT

vendor,

vendor\_name,

AVG(bottle\_price::numeric)::money AS average\_bottle\_price

FROM products

GROUP BY vendor\_name, vendor

HAVING AVG(bottle\_price::numeric) < 100

ORDER BY 3 DESC

LIMIT 100;

SELECT \* FROM products;

SELECT

vendor\_name,

MIN(bottle\_price::numeric) AS min\_bottle\_price,

MAX(bottle\_price::numeric) AS max\_bottle\_price

FROM products

GROUP BY vendor\_name

HAVING MIN(bottle\_price::numeric) = 10

ORDER BY 3

LIMIT 20;

SELECT CONCAT(item, ' - ', description) AS item\_description

FROM sales

LIMIT 100;

SELECT REPLACE(description, 'Absolut', 'Grey Goose')

FROM sales

LIMIT 100;

SELECT

description,

LENGTH(description) - LENGTH(REPLACE(description, ' ', '')) + 1 AS word\_count

FROM sales

LIMIT 100;

SELECT

UPPER(description),

LOWER(description)

FROM sales

LIMIT 100;

SELECT LEFT(description, 5)

FROM sales

LIMIT 100;

SELECT SUBSTRING(description, 3, 10)

FROM sales

LIMIT 100;

-- SELECT TRIM(LEADING '0' FROM ...)

SELECT (btl\_price - state\_btl\_cost)/btl\_price AS percentage\_price\_difference

FROM sales

LIMIT 100;

SELECT ROUND((proof::numeric / 200) \* bottle\_size, 2)

FROM products

LIMIT 100