# Summarizing Data with SQL

## Summary Statistics

32) How many rows are in the `pets` table?

13 - SELECT \* FROM pets;

33) How many female pets are in the `pets` table?

7 - SELECT \* FROM pets

WHERE sex = 'F';

34) How many female cats are in the `pets` table?

4 - SELECT COUNT(\*) FROM pets

WHERE sex = 'F' AND species = 'cat';

35) What's the mean age of pets in the `pets` table?

5 - SELECT AVG(age) FROM pets;

36) What's the mean age of dogs in the `pets` table?

6.5 - SELECT AVG(age) FROM pets

WHERE species = 'dog';

37) What's the mean age of male dogs in the `pets` table?

8 - SELECT AVG(age) FROM pets

WHERE species = 'dog' AND sex = 'M';

38) What's the count, mean, minimum, and maximum of pet ages in the `pets` table?

\* \_NOTE:\_ SQLite doesn't have built-in formulas for standard deviation or median!

COUNT = 13, Mean = 5.2, Min = 1, Max = 10

SELECT “AGGREGATE FUNCTION”(age) FROM pets;

39) Repeat the previous problem with the following stipulations:

\* Round the average to one decimal place.

Count = 13, Mean = 5, Min = 1, MAX = 10

SELECT ROUND(“AGGREGATE FUNCTION”(age)) FROM pets;

\* Give each column a human-readable column name (for example, "Average Age")

SELECT ROUND(COUNT(age)) AS "Age Count" FROM pets;

SELECT ROUND(AVG(age)) AS "Average Age" FROM pets;

SELECT ROUND(MIN(age)) AS "Minimum Age" FROM pets;

SELECT ROUND(MAX(age)) AS "Maximum Age" FROM pets;

40) How many rows in `employees\_null` have missing salaries?

10 - SELECT \* FROM employees\_null

WHERE salary IS NULL;

41) How many salespeople in `employees\_null` having \_nonmissing\_ salaries?

60 - SELECT \* FROM employees\_null

WHERE job = 'Sales' and salary IS NOT NULL;

42) What's the mean salary of employees who joined the company after 2010? Go back to the usual `employees` table for this one.

\* \_Hint:\_ You may need to use the `CAST()` function for this. To cast a string as a float, you can do `CAST(x AS REAL)`

$78,070.

SELECT ROUND(AVG(salary)) FROM employees

WHERE salary > CAST(2010-01-01 AS REAL);

43) What's the mean salary of employees in Swiss Francs?

\* \_Hint:\_ Swiss Francs are abbreviated "CHF" and 1 USD = 0.97 CHF.

75,728CHF SELECT ROUND(AVG(salary \* 0.97)) AS CHF FROM employees;

44) Create a query that computes the mean salary in USD as well as CHF. Give the columns human-readable names (for example "Mean Salary in USD"). Also, format them with comma delimiters and currency symbols.

\* \_NOTE:\_ Comma-delimiting numbers is only available for integers in SQLite, so rounding (down) to the nearest dollar or franc will be done for us.

\* \_NOTE2:\_ The symbols for francs is simply `Fr.` or `fr.`. So an example output will look like `100,000 Fr.`.

SELECT to\_char((AVG(salary \* 0.97)), 'fm999,999,999 FR') AS Mean\_Salary\_CHF,

to\_char((AVG(salary)), 'fm$999,999,999') AS Mean\_Salary\_USD

FROM employees;

## Aggregating Statistics with GROUP BY

45) What is the average age of `pets` by species?

Dog - 7, Lobster - 3, Cat - 4

SELECT ROUND(AVG(age)) FROM pets

GROUP BY species;

46) Repeat the previous problem but make sure the species label is also displayed! Assume this behavior is always being asked of you any time you use `GROUP BY`.

SELECT ROUND(AVG(age)), species FROM pets

GROUP BY species;

47) What is the count, mean, minimum, and maximum age by species in `pets`?

Mean: Dog - 7, Lobster - 3, Cat - 4

Count: Dog - 6, Lobster - 1, Cat – 6

Minimum: Dog - 1, Lobster - 3, Cat - 2

Maximum: Dog - 10, Lobster - 3, Cat – 7

SELECT ROUND(“AGGREGATE FUNCTION”(age)), species FROM pets

GROUP BY species;

48) Show the mean salaries of each job title in `employees`.

SELECT ROUND(AVG(salary)), job FROM employees

GROUP BY job;

49) Show the mean salaries in New Zealand dollars of each job title in `employees`.

\* \_NOTE:\_ 1 USD = 1.65 NZD.

SELECT to\_char(ROUND((AVG(salary \* 1.65))), 'fm999,999,999' NZ) AS NZD, job FROM employees

GROUP BY job;

50) Show the mean, min, and max salaries of each job title in `employees`, as well as the numbers of employees in each category.

SELECT ROUND(AVG(salary)), MIN(salary), MAX(salary), COUNT(job), job FROM employees

GROUP BY job;

51) Show the mean salaries of each job title in `employees` sorted descending by salary.

SELECT ROUND(AVG(salary)), job FROM employees

GROUP BY job

ORDER BY AVG(salary) DESC;

52) What are the top 5 most common first names among `employees`?

SELECT firstname, COUNT(firstname) FROM employees

GROUP BY firstname

ORDER BY COUNT(firstname) DESC

LIMIT 5;

53) Show all first names which have exactly 2 occurrences in `employees`.

SELECT firstname, COUNT(firstname) FROM employees

GROUP BY firstname

HAVING COUNT(firstname) = 2;

54) Take a look at the `transactions` table to get an idea of what it contains. Note that a transaction may span multiple rows if different items are purchased as part of the same order. The employee who made the order is also given by their ID.

SELECT \* FROM transactions;

55) Show the top 5 largest orders (and their respective customer) in terms of the numbers of items purchased in that order.

SELECT order\_id, SUM(quantity), customer FROM transactions

GROUP BY customer, order\_id

ORDER BY SUM(quantity) DESC

LIMIT 5;

56) Show the total cost of each transaction.

\* \_Hint:\_ The `unit\_price` column is the price of \_one\_ item. The customer may have purchased multiple.

\* \_Hint2:\_ Note that transactions here span multiple rows if different items are purchased.

SELECT order\_id, SUM(unit\_price \* quantity) AS total\_cost FROM transactions

GROUP BY order\_id

ORDER BY order\_id;

57) Show the top 5 transactions in terms of total cost.

SELECT order\_id, SUM(unit\_price \* quantity) AS total\_cost FROM transactions

GROUP BY order\_id

ORDER BY total\_cost DESC

LIMIT 5;

58) Show the top 5 customers in terms of total revenue (ie, which customers have we done the most business with in terms of money?)

SELECT customer, SUM(unit\_price \* quantity) AS total\_cost FROM transactions

GROUP BY customer

ORDER BY total\_cost DESC

LIMIT 5;

59) Show the top 5 employees in terms of revenue generated (ie, which employees made the most in sales?)

SELECT employee\_id, SUM(unit\_price \* quantity) AS total\_cost FROM transactions

GROUP BY employee\_id

ORDER BY total\_cost DESC

LIMIT 5;

60) Which customer worked with the largest number of employees?

\* \_Hint:\_ This is a tough one! Check out the `DISTINCT` keyword.

Kelly-Wright – 61 employees

SELECT customer, COUNT(DISTINCT employee\_id) FROM transactions

GROUP BY customer

ORDER BY Count(DISTINCT employee\_id) DESC;

61) Show all customers who've done more than $80,000 worth of business with us.

SELECT customer, SUM(unit\_price \* quantity) AS total\_cost FROM transactions

GROUP BY customer

HAVING SUM(unit\_price \* quantity) > 80000

ORDER BY total\_cost DESC;