Aggregation

Announcements

Select Statements

Grouping Rows

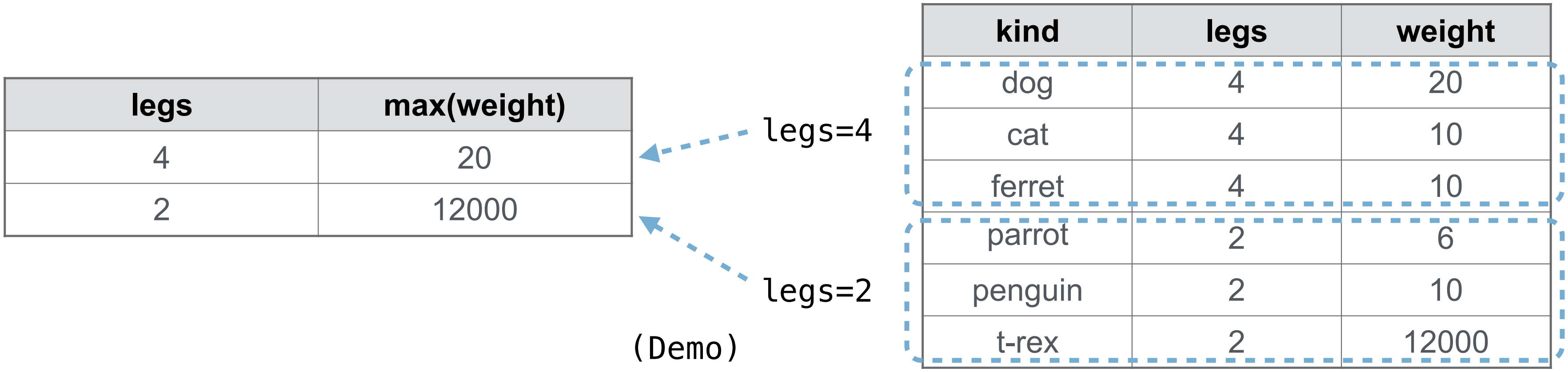
Rows in a table can be grouped, and aggregation is performed on each group

```
[expression] AS [name], [expression] AS [name], ...
```

```
SELECT [columns] FROM [table] GROUP BY [expression] HAVING [expression];
```

The number of groups is the number of unique values of an expression

```
SELECT legs, MAX(weight) FROM animals GROUP BY legs;
```



legs	max(weight)
4	20
2	12000

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

(Demo)

Writing Select Statements

Describe the output table:

1) Determine which existing rows are needed to express the result (FROM & WHERE)

2) Form groups and determine which groups should appear as output rows (GROUP BY & HAVING)

3) Format the output rows (SELECT)

SELECT: Values each output row contains (and column labels)

FROM: Source of input rows

WHERE: Which input rows

GROUP BY: Form output rows

HAVING: Which output rows

Example: UC Berkeley Employee Counts

```
(Demo)
```

Example: Select Statement Components

```
For each type of employee, compute the fa23-fa18 difference in the total headcount,
but include a row only for each type for which the headcount increased.
```

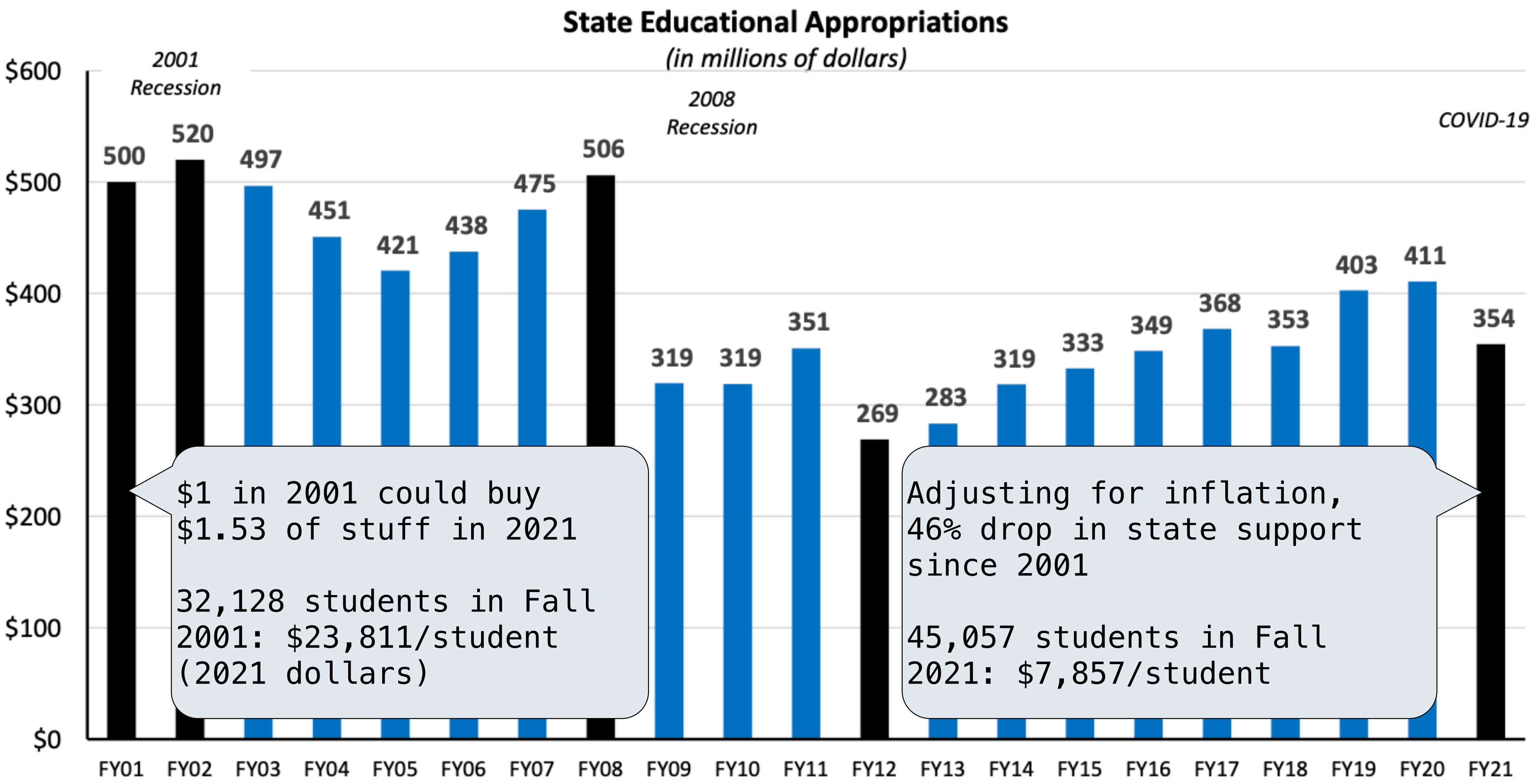
```
sqlite> SELECT * FROM cal;
source    type                 role                             fa08   fa13   fa18   fa23
--------  -------------------  -------------------------------  -----  -----  -----  -----
employee  Grad Student Titles  Grad St. Instructor (GSI)        1943   1925   2202   2248
                ...                                        ...    ...    ...    ...
student   Grad Student         Grad Student                     10258  10253  11666  12621
student   Undergrad            Undergrad                        25151  25951  30853  33078
```

SELECT: Values each output row contains (and column labels)

```
SELECT type, SUM(fa23) - SUM(fa18) AS increase
  FROM cal                            FROM: Source of input rows
  WHERE source = "employee"           WHERE: Which input rows
  GROUP BY type                       GROUP BY: Form output rows
  HAVING SUM(fa23) > SUM(fa18);       HAVING: Which output rows
```

```
type                 increase
-------------------  --------
Grad Student Titles  327
Other Faculty        352
Regular Faculty      48
Staff                454
```

The University is a public institution, so it is supported to an extent by California taxpayers through an allocation by the state government. In the past, generous state support allowed UC Berkeley to operate while keeping costs to students low. While still an important revenue source, the state's financial support of the university has diminished significantly. Thirty years ago, 50 percent of the university's revenue came from the state, but today, the state provides just 14 percent of the university's revenue.



$1 in 2001 could buy
$1.53 of stuff in 2021

32,128 students in Fall
2001: $23,811/student
(2021 dollars)

Adjusting for inflation,
46% drop in state support
since 2001

45,057 students in Fall
2021: $7,857/student

Joins Practice

Discussion Question

What's the maximum difference between leg count for two animals with the same weight?

Approach #1: Consider all pairs of animals.

```
SELECT MAX(a.legs - b.legs) AS difference
  FROM animals AS a, animals AS b
  WHERE a.weight = b.weight;
```

Approach #2: Group by weight.

```
SELECT MAX(legs) - MIN(legs) AS difference
   FROM animals
   GROUP BY weight
   ORDER BY difference DESC
   LIMIT 1;
```

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

difference
2

Discussion Question

What are all the kinds of animals that have the maximal number of legs?

```
sqlite> SELECT * FROM animals WHERE legs = MAX(legs);
Parse error: misuse of aggregate function MAX()
```

animals:

kind	legs	weight
dog	4	20
cat	4	10
ferret	4	10
parrot	2	6
penguin	2	10
t-rex	2	12000

Approach #1: Give the maximum number of legs a name.

```
CREATE TABLE m AS SELECT MAX(legs) AS max_legs FROM animals;
SELECT kind FROM animals, m WHERE legs = max_legs;
```

Approach #2: For each kind of animal, compare its legs to the maximum legs by grouping.

```
SELECT a.kind FROM animals AS a, animals AS b GROUP BY a.kind HAVING a.legs = MAX(b.legs);
```

Group By Practice

Spring 2023 CS 61A Final Question 7

The finals table has columns hall (strings) and course (strings), and has rows for each lecture hall in which a course is holding its final exam.

The sizes table has columns room (strings) and seats (numbers), and has one row per unique room on campus containing the number of seats in that room. All lecture halls are rooms.

Create a table with two columns, course (string) and seats (number), and with one row containing the **name of the course** and the **total number of seats in final rooms** for that course. Only include a row **for each course that uses at least two rooms for its final.**

finals:

hall	course
RSF	61A
Wheeler	61A
RSF	61B

sizes:

room	seats
RSF	900
Wheeler	700
310 Soda	40

result:

course	seats
61A	1600

```
SELECT course, SUM(seats) AS seats
   FROM finals, sizes
   WHERE hall=room
   GROUP BY course
   HAVING COUNT(*) > 1 ;
```