Challenge 8: Gradebook

Wednesday, April 5, 2017 11:45 AM

Model

```
CREATE TABLE exercise_logs
   (id INTEGER PRIMARY KEY AUTOINCREMENT,
   type TEXT,
   minutes INTEGER,
   calories INTEGER,
   heart_rate INTEGER);


INSERT INTO exercise_logs(type, minutes, calories, heart_rate) VALUES ("hiking", 60, 80, 85);

SELECT * FROM exercise_logs;

SELECT COUNT(*) FROM exercise_logs WHERE heart_rate > 220 - 30;

/* 50-90% of max*/
SELECT COUNT(*) FROM exercise_logs WHERE
   heart_rate >= ROUND(0.50 * (220-30))
   AND  heart_rate <= ROUND(0.90 * (220-30));

/* CASE */
SELECT type, heart_rate,
   CASE
      WHEN heart_rate > 220-30 THEN "above max"
      WHEN heart_rate > ROUND(0.90 * (220-30)) THEN "above target"
      WHEN heart_rate > ROUND(0.50 * (220-30)) THEN "within target"
      ELSE "below target"
   END as "hr_zone"
FROM exercise_logs;

SELECT COUNT(*),
   CASE
      WHEN heart_rate > 220-30 THEN "above max"
      WHEN heart_rate > ROUND(0.90 * (220-30)) THEN "above target"
      WHEN heart_rate > ROUND(0.50 * (220-30)) THEN "within target"
      ELSE "below target"
   END as "hr_zone"
FROM exercise_logs
GROUP BY hr_zone;
```

Step 1

We've created a database to track student grades, with their name, number grade, and what percent of activities they've completed. In this first step, select all of the rows, and display the name, number_grade, and percent_completed, which you can compute by multiplying and rounding the fraction_completed column.

```
SELECT name, number_grade, ROUND(fraction_completed * 100) AS percent_completed
FROM student_grades;
```

DATABASE SCHEMA

student_grades
6 rows

id (PK)INTEGER
nameTEXT
number_gradeINTEGER
fraction_completedREAL

RESULTS

name	number_grade	percent_completed
Winston	90	81
Winnefer	95	90
Winsteen	85	91
Wincifer	66	71
Winster	76	50
Winstonia	82	90

Step 2
Now, this step is a little tricky. The goal is a table that shows how many students have earned which letter_grade. You can output the letter_grade by using CASE with the number_grade column, outputting 'A' for grades > 90, 'B' for grades > 80, 'C' for grades > 70, and 'F' otherwise. Then you can use COUNT with GROUP BY to show the number of students with each of those grades.

```
SELECT COUNT(*),
CASE
WHEN number_grade > 90  THEN "A"
      WHEN number_grade > 80  THEN "B"
      WHEN number_grade > 70 THEN "C"
      ELSE "F"
   END as "letter_grade"
FROM student_grades
GROUP BY letter_grade;
```

DATABASE SCHEMA

student_grades
6 rows

id (PK)INTEGER
nameTEXT
number_gradeINTEGER
fraction_completedREAL

RESULTS

name	number_grade	percent_completed
Winston	90	81
Winnefer	95	90
Winsteen	85	91
Wincifer	66	71
Winster	76	50
Winstonia	82	90

COUNT(*)	letter_grade
1	A
3	B
1	C
1	F