Full Stack Development/Software Workshop 2

Formative Exercises 3

Based on lectures: 7

Introduction

Exercises 1 to 5 below are the same for 'Formative Exercises 2'. This time use a **subquery** in your solution.

Use subqueries in your solutions to all of these exercises.

Exercise 1

Write a query that lists all of the distinct *custids* of customers who bought albums on the *label* 'RCA'.

Exercise 2

Write a query that lists all of the postcodes of customers who have reviewed an album.

Exercise 3

Write a query that lists all of the postcodes of customers who have bought an album and reviewed an album (any album, not necessarily one they bought).

Think about how many joins you will need for this, and what type.

Exercise 4

Write a query that lists all of the postcodes of customers who have bought an album and reviewed **that** album.

Exercise 5

Write a query to list the *title*, *year*, *price*, *genre* and genre *description* of albums released after 1970 that have at least one sale. Order the result by year in descending order.

Exercise 6

Write a query to find the amount of money spent by customer Jimmy Osmond.

The result should contain one row and one column: the amount spent.

Exercise 7

Write a query to find the titles of albums that have been reviewed but not bought by anyone.

Exercise 8

Write a query to find the genre that has been reviewed most often.

In the result show the genre and the number of times it has been reviewed.

Hint: This query can be formulated in a number of ways. One way to think about this is that you want the count of reviews for a genre that is greater than or equal to the count of reviews for all genres. Have a look at the SQL **ALL** operator online to see if you can make use of that.