

# Querying Relational Databases Part 6 Notes

Team Treehouse

June 8, 2020

## 1 What are Subqueries

- is used to return data that will be used in the main query as a condition
- is embedded within the WHERE clause
- Use cases
  - Criteria for a where clause is not specifically known
  - Need a temporary dataset to join with other tables in your database

## 2 Using IN with Subqueries to Filter Data

- **Syntax:** SELECT *columns name* FROM *table 1 name* WHERE *Column 1 name* IN (SELECT *Column 1 Name* FROM *table 2 name* WHERE *Search Criteria*)
- **Syntax 2:** SELECT *columns name* FROM *table 1 name* WHERE *Column 1 name* NOT IN (SELECT *Column 1 Name* FROM *table 2 name* WHERE *Search Criteria*)

### Example:

```
1 SELECT * FROM Sale WHERE CarID NOT IN (SELECT CarID FROM Car WHERE
2 ModelYear = 2015);
```

### 3 Using a Subquery to Create a Temporary Table (Part 1)

- Temporary table is also called a **derived table**
- Is used in INNER JOIN and OUTER JOIN
- **Syntax:**

*SELECT Columns Name*

*FROM Table name*

*INNER / OUTER JOIN*

*(SELECT columns name FROM table name WHERE Search Criteria) AS Alias*

*ON Join Criteria;*

#### Example:

```
1  SELECT * FROM Sale AS s
2      INNER JOIN (SELECT CarID FROM Car WHERE ModelYear = 2015) AS t
3      ON s.CarID = t.CarID;
4
```

### 4 Using a Subquery to Create a Temporary Table (Part 2)

- Note: Always work on subqueries before working more complex query
- **Derived table** or **temporary table** can also be used in JOINS

```
1  // ===== create subqueries =====
2
3  SELECT SalesRepID, SUM(SaleAmount) AS StLouisAmount
4  FROM Sales AS s WHERE s.LocationID = 1
5  GROUP BY SalesRepID;
6
7
8  SELECT SalesRepID, SUM(SaleAmount) AS ColumbiaAmount
9  FROM Sales AS s WHERE s.LocationID = 2
10 GROUP BY SalesRepID;
11
12 // ===== Add to main query =====
13
14 SELECT sr.LastName FROM SalesRep AS sr
```

```
15     LEFT OUTER JOIN (  
16         SELECT SalesRepID, SUM(SaleAmount) AS StLouisAmount  
17         FROM Sale AS s WHERE s.LocationID = 1  
18         GROUP BY SalesRepID;  
19     ) AS Loc1  
20  
21     LEFT OUTER JOIN (  
22         SELECT SalesRepID, SUM(SaleAmount) AS ColumbiaAmount  
23         FROM Sale AS s WHERE s.LocationID = 2  
24         GROUP BY SalesRepID;  
25     ) AS Loc2  
26  
27  
28     // ===== Finish by adding join criteria =====  
29  
30     SELECT sr.LastName, Loc1.StLouisAmount, Loc2.ColumbiaAmount FROM  
31     SalesRep AS sr  
32     LEFT OUTER JOIN (  
33         SELECT SalesRepID, SUM(SaleAmount) AS StLouisAmount  
34         FROM Sale AS s WHERE s.LocationID = 1  
35         GROUP BY SalesRepID;  
36     ) AS Loc1 ON sr.SalesRepID = Loc1.SalesRepID  
37  
38     LEFT OUTER JOIN (  
39         SELECT SalesRepID, SUM(SaleAmount) AS ColumbiaAmount  
40         FROM Sale AS s WHERE s.LocationID = 2  
41         GROUP BY SalesRepID;  
42     ) AS Loc2 ON sr.SalesRepID = Loc2.SalesRepID  
43  
44
```

## 5 Subqueries Review

1. When using a subquery to filter your outer query, you use what SQL keyword?
  - A. AS
  - B. IN
  - C. OF
  - D. ON

**Answer:** B

2. Why must a derived table be aliased?
  - A. The database wouldn't be able to tell which user is running the query otherwise.

- B. Being a temporary object, it has no other name and thus the database wouldn't be able to reference its resulting data set.
- C. They do not need to be aliased.

**Answer: B**

3. When creating derived / temporary tables, your subquery belongs in which section of the query:
- A. WHERE
  - B. GROUP BY
  - C. FROM
  - D. SELECT

**Answer: C**

4. When using a subquery to create a derived table, you can only select ONE column in the Subquery.
- A. True
  - B. False

**Answer: B**

5. When using IN with a subquery, you can only select ONE column in the Subquery.
- A. True
  - B. False

**Answer: A**

## 6 Review and Practice

### 7 Exercise 1

- Solution included in *exercise\_1.sql*