Database Programming with SQL

10-1
Fundamentals of Subqueries
Objectives

This lesson covers the following objectives:

• Define and explain the purpose of subqueries for retrieving data

• Construct and execute a single-row subquery in the WHERE clause

• Distinguish between single-row and multiple-row subqueries
Purpose

• Has a friend asked you to go to a movie, but before you could answer "yes" or "no", you first had to check with your parents?

• Has someone asked you the answer to a math problem, but before you can give the answer, you had to do the problem yourself?

• Asking parents, or doing the math problem, are examples of subqueries.

• In SQL, subqueries enable us to find the information we need so that we can get the information we want.
Subquery Overview

• Throughout this course, you have written queries to extract data from a database.

• What if you wanted to write a query, only to find out you didn't have all the information you needed to construct it?

• You can solve this problem by nesting queries—placing one query inside the other query.

• The inner query is called a "subquery."
Subquery Overview

• The subquery executes to find the information you don't know.

• The outer query uses that information to find out what you need to know.

• Being able to combine two queries into one can be very useful when you need to select rows from a table with a condition that depends on the data in the table itself.
Subquery Overview

- A subquery is a SELECT statement that is embedded in a clause of another SELECT statement.
- A subquery executes once before the main query.
- The result of the subquery is used by the main or outer query.
- Subqueries can be placed in a number of SQL clauses, including the WHERE clause, the HAVING clause, and the FROM clause.
- The subquery syntax is:

  ```sql
  SELECT select_list
  FROM table
  WHERE expression operator
  (SELECT select_list
   FROM table);
  ```

  The SELECT statement in parentheses is the inner query or 'subquery'. It executes first, before the outer query.
Guidelines for Using Subqueries

• Guidelines:
  – The subquery is enclosed in parentheses.
  – The subquery is placed on the right side of the comparison condition.
  – The outer and inner queries can get data from different tables.
  – Only one ORDER BY clause can be used for a SELECT statement; if used, it must be the last clause in the outer query.
  – A subquery cannot have its own ORDER BY clause.
  – The only limit on the number of subqueries is the buffer size the query uses.
Two Types of Subqueries

- The two types of subqueries are:
  - Single-row subqueries that use single-row operators (> =, >=, <, <> <=) and return only one row from the inner query.
  - Multiple-row subqueries that use multiple-row operators (IN, ANY, ALL) and return more than one row from the inner query.
Subquery Example

• What if you wanted to find out the names of the employees that were hired after Peter Vargas?

• The first thing you need to know is the answer to the question, "When was Peter Vargas hired?"

• Once you know his hire date, then you can select those employees whose hire dates are after his.

SELECT first_name, last_name, hire_date
FROM employees
WHERE hire_date >
  (SELECT hire_date
   FROM employees
   WHERE last_name = 'Vargas');

<table>
<thead>
<tr>
<th>FIRST_NAME</th>
<th>LAST_NAME</th>
<th>HIRE_DATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Eleni</td>
<td>Zlotkey</td>
<td>29/Jan/2000</td>
</tr>
<tr>
<td>Kimberely</td>
<td>Grant</td>
<td>24/May/1999</td>
</tr>
<tr>
<td>Kevin</td>
<td>Mourgos</td>
<td>16/Nov/1999</td>
</tr>
<tr>
<td>Diana</td>
<td>Lorentz</td>
<td>07/Feb/1999</td>
</tr>
</tbody>
</table>
Subquery and Null

• If a subquery returns a null value or no rows, the outer query takes the results of the subquery (null) and uses this result in its WHERE clause.

• The outer query will then return no rows, because comparing any value with a null always yields a null.

```
SELECT last_name
FROM employees
WHERE department_id =
    (SELECT department_id
     FROM employees
     WHERE last_name = 'Grant');
```

no data found
Subquery and Null

• Who works in the same department as Grant?
• Grant's department_id is null, so the subquery returns NULL.
• The outer query then substitutes this value in the WHERE clause (WHERE department_id = NULL).
• The outer query returns no rows, because comparing anything with a null returns a null.

```
SELECT last_name
FROM employees
WHERE department_id =
  (SELECT department_id
   FROM employees
   WHERE last_name = 'Grant');
```

no data found
Terminology

Key terms used in this lesson included:

• Subquery
• Inner query
• Outer subquery
• Single-row subquery
• Multiple-row subquery
Summary

In this lesson, you should have learned how to:

• Define and explain the purpose of subqueries for retrieving data
• Construct and execute a single-row subquery in the WHERE clause
• Distinguish between single-row and multiple-row subqueries