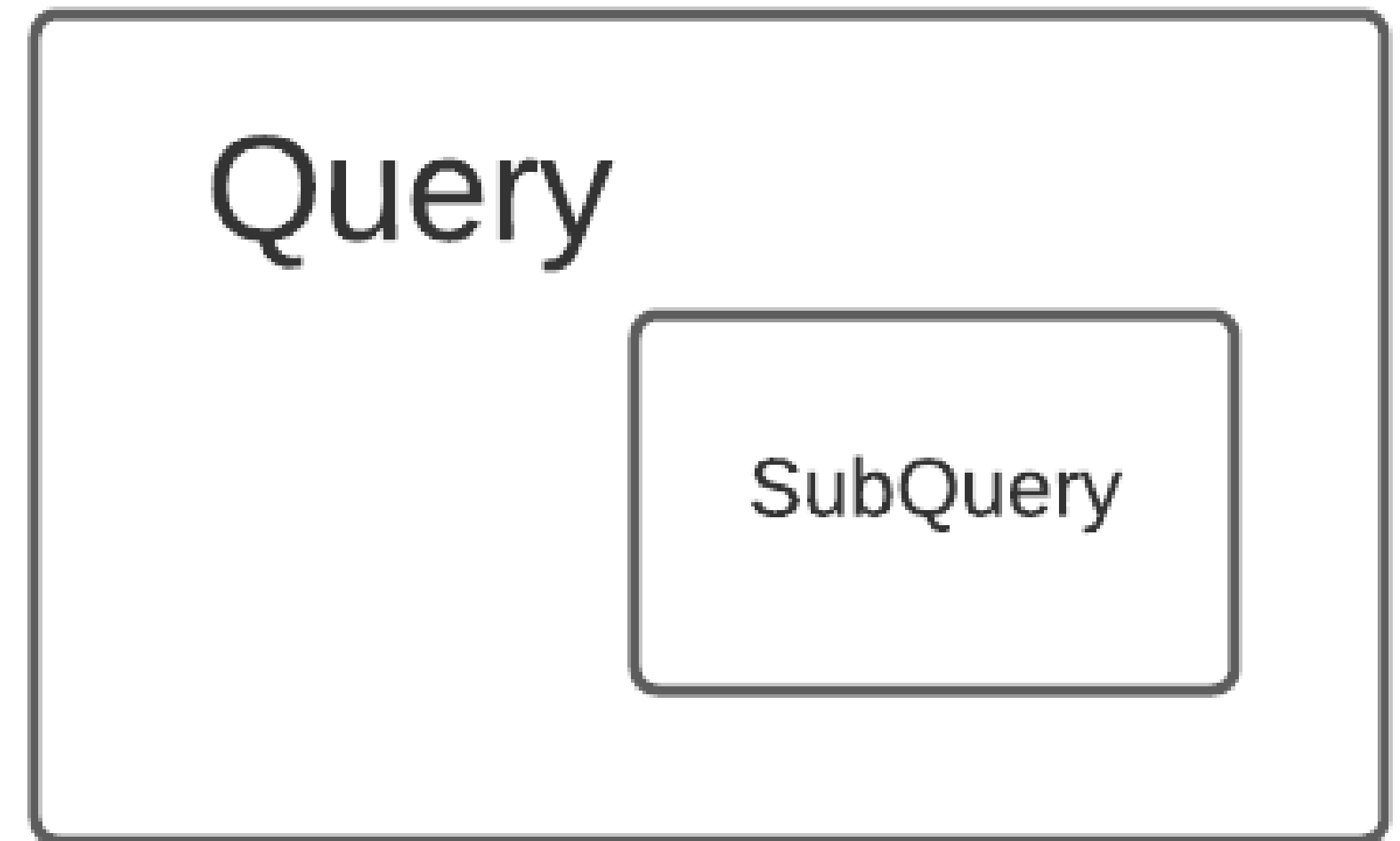


SQL SubQueries

The SubQuery Simply Defined:

A query within a query



SQL SubQueries

SubQuery – Three modes of use

1. Subquery in WHERE.

The answer set to the “inner” query is used as a predicate in a where clause in the “outer” query.

2. Subquery in SELECT.

The answer set to the “inner” query is used as a column in a select

3. Subquery in FROM.

The answer set to the “inner” query is used as a "virtual" table in a from clause.

Subquery must always be enclosed in parentheses!

SQL SubQueries

Example 1 – Predicate in a WHERE clause with = operator

Find the productID, name and unit price of the highest priced product Northwinds sells.

```
select productID, productname, unitprice
  from "alanparadise/nw"."products"
 where UnitPrice = (
    select MAX(UnitPrice)
      from "alanparadise/nw"."products" )
```

Note that with the “equals” condition, the inner query returns only one value (one row, one column)

SQL SubQueries

Example 2 – Predicate in a WHERE clause with "in" operator

Find the CustomerID and the OrderID for all orders with more than 100 units sold.

```
select customerID, orderID
  from "alanparadise/nw"."orders"
 where orderID in (

      select orderID
        from "alanparadise/nw"."orderdetails"
       where quantity > 100 )
```

Order by CustomerID

Note that with the “in” condition, the inner query returns many values (many rows, one column) as a list

NOTE also: this is hitting TWO different tables

SQL SubQueries

Example 3 – Expression in a SELECT clause

List each product name and the total value of that product's orders

```
select productname, (select sum(UnitPrice * quantity)
                     from "alanparadise/nw"."orderdetails"
                     where "alanparadise/nw"."orderdetails".productid =
                           "alanparadise/nw"."products".productid) as "Total"
from "alanparadise/nw"."products"
```

Note that the inner query returns only one value (one row, one column)

SQL SubQueries

Example 4 – Subquery in FROM.

Create a list of all orders with fewer than 100 items sold.

```
SELECT orderID
FROM (SELECT orderID, SUM(quantity)
      from "alanparadise/nw"."orderdetails"
      GROUP BY OrderID
      HAVING SUM(quantity) < 100) AS DetailCount;
```

NOTE: the subquery must have an alias name

SQL SubQueries

Co-Related SubQuery

Special type of subquery

- The inner query references a value from the outer query
- The inner query is executed each time the outer query returns another row.

NOTE: Can yield performance issues.

SQL SubQueries

Example 5 – Correlated Sub-Query

Select all the employees where the employee had orders shipped to customers in the employee's own home city.

```
SELECT O.employeeID, orderID, shipcity, customerID
FROM "alanparadise/nw"."orders" O
WHERE employeeID IN
(SELECT employeeID FROM "alanparadise/nw"."employees" E
WHERE E.City = O.shipcity);
```


SQL SubQueries

NOTE: This example uses a temporary alias for the table name.

```
FROM "alanparadise/nw"."employees" E
```



This serves two purposes:

1. The employeeid is ambiguous, since it appears in both the orders table and the employees table.
2. The alias saves typing and makes the query code more clear and readable

SQL Subqueries – Lab # 5

This concludes Module 3, Lesson 3, "Subqueries"

Next step: Follow the instructions for Lab # 5