

# TECH

on the Net

[Download](#)[Play Now](#)[Home](#)   [Feedback](#)[Site Map](#) [Microsoft  
Access](#)[Excel  
Word](#)[Database  
SQL  
Oracle / PLSQL](#)[Languages  
C Language](#)[UNIX  
General UNIX  
Linux](#)[Misc  
ASCII Table  
Java  
Clipart](#)[Other Sites  
CheckYourMath  
BigActivities](#)[Home](#) > [Oracle/PLSQL](#)

## Oracle/PLSQL: Subqueries

### What is a subquery?

A subquery is a query within a query. In Oracle, you can create subqueries within your SQL statements. These subqueries can reside in the WHERE clause, the FROM clause, or the SELECT clause.

### WHERE clause

Most often, the subquery will be found in the WHERE clause. These subqueries are also called **nested subqueries**.

For example:

```
select * from all_tables tabs
where tabs.table_name in (select cols.table_name
                        from all_tab_columns cols
                        where cols.column_name = 'SUPPLIER_ID');
```

### Limitations

Oracle allows up to 255 levels of subqueries in the WHERE clause.

### FROM clause

A subquery can also be found in the FROM clause. These are called **inline views**.

For example:

```
select suppliers.name, subquery1.total_amt
from suppliers,
(select supplier_id, Sum(orders.amount) as total_amt
 from orders
 group by supplier_id) subquery1,
where subquery1.supplier_id = suppliers.supplier_id;
```

In this example, we've created a subquery in the FROM clause as follows:

```
(select supplier_id, Sum(orders.amount) as total_amt
 from orders
 group by supplier_id) subquery1
```

This subquery has been aliased with the name *subquery1*. This will be the name used to reference this subquery or any of its fields.

### Limitations

Oracle allows an unlimited number of subqueries in the FROM clause.

### SELECT clause

A subquery can also be found in the SELECT clause.

For example:

```
select tbs.owner, tbs.table_name,
(select count(column_name) as total_columns
 from all_tab_columns cols
 where cols.owner = tbs.owner
 and cols.table_name = tbs.table_name) subquery2
from all_tables tbs;
```



In this example, we've created a subquery in the SELECT clause as follows:

```
(select count(column_name) as total_columns
 from all_tab_columns cols
 where cols.owner = tbls.owner
 and cols.table_name = tbls.table_name) subquery2
```

The subquery has been aliased with the name *subquery2*. This will be the name used to reference this subquery or any of its fields.

The trick to placing a subquery in the select clause is that the subquery must return a single value. This is why an aggregate function such as **SUM**, **COUNT**, **MIN**, or **MAX** is commonly used in the subquery.

---

[Terms of Service](#)   [Privacy Policy](#)

**Copyright** © 2003-2013 TechOnTheNet.com. All rights reserved.  
We are not responsible for any loss or liability incurred by using this information.

---