**Correlated Subqueries versus Inner Joins**

It is important to understand that you can get that same results using either a subquery or join.  Though both return the same results, there are advantages and disadvantages to each method!

Consider the last example where we count line items for SalesHeader items.

SELECT SalesOrderID,

OrderDate,

TotalDue,

(SELECT COUNT(SalesOrderDetailID)

FROM Sales.SalesOrderDetail

WHERE SalesOrderID = SO.SalesOrderID) as LineCount

FROM  Sales.SalesOrderHeader SO

This same query can be done using and INNER JOIN along with GROUP BY as

SELECT   SO.SalesOrderID,

OrderDate,

TotalDue,

COUNT(SOD.SalesOrderDetailID) as LineCount

FROM     Sales.SalesOrderHeader SO

INNER JOIN Sales.SalesOrderDetail SOD

ON SOD.SalesOrderID = SO.SalesOrderID

GROUP BY SO.SalesOrderID, OrderDate, TotalDue

* Usually joins will work faster than inner queries, but in reality it will depend on the execution plan generated by SQL Server. No matter how you write your query, SQL Server will always transform it on an execution plan. If it is "smart" enough to generate the same plan from both queries, you will get the same result.
* Subquery makes for busy disk access, think of hard disk's read-write needle(head?)
* Join works by concentrating the operation on the result of the first two tables, any subsequent joins would concentrate joining on the in-memory(or cached to disk) result of the first joined tables, and so on. less read-write needle movement, thus faster