**\*\* T-SQL JOIN 과 ANIS-SQL JOIN의 결과가 다른 이유는?**

MS-SQL 에서는 **T-SQL** 및 **ANSI-SQL**을 이용하여 조인문을 사용할수 있다
대부분의 경우에는 각쿼리는 동일하며 올바른 결과를 생성한다
그러나 외부결합을 WHERE 절에서 IS NULL 과 결부시켰을 경우에는 문제가 발행한다

Northwind 의 샘플 데이터베이스에 대한 두 쿼리를 보자!

**[T-SQL LEFT OUTER JOIN]**

SELECT c.customerid
FROM customers c, orders o
WHERE c.customerid \*= o.customerid
           AND o.customerid IS NULL
ORDER BY o.customerid



결과 : 91행의 결과

**[ANSI-SQL LEFT OUTER JOIN]**

SELECT c.customerid
FROM customers c LEFT OUTER JOIN  orders o ON c.customerid = o.customerid
WHERE  o.customerid IS NULL
ORDER BY o.customerid



결과 : 2행의 결과

첫번째 T-SQL 이 왜 잘못된 결과 집합을 리턴 했는지 알아보자

이 쿼리의 목적은 customers 테이블을 orders 테이블에 결합시킨뒤

orders 테이블에서 customerid 값이 NULL 인 레코드를 찾아내는 것이다

하지만 2행이라는 옳은 결과대신 91행이라는 부정확한 결과를 얻은 이유는

**OUTER JOIN 조건문이 FROM 절 대신에 WHERE 절에 놓여져서 SQL서버가**

**o.customerid에 대한 IS NULL 검사를 결합이전에 이행하기 때문이다**

orders 테이블에 있는 어떤 행도 NULL customerid 값을 가지고 있지 않으므로 SQL 서버는

와부 결합을 처리하기 위하여 이전에 Orders 테이블의 모든 행을 제거한다.

그래서 SQL  서버가 결합을 처리하려고 할때에는 orders 테이블에서 어떤 행도 발견할수 없어

모든 고객이 주분이 없은 것으로 부정확하게 리턴 되는 것이다.

ANSI 구문을 사용하게 되면 SQL 서버는 모든 경우에 있어 JOIN표현을 WHERE 조건전에 검토한다.

JOIN조건 문들을 항상 WHERE 절 대신에 FROM 절에 놓기를 권한다.

외부결합을 할때에는 특히나 그러한데 위와 같은 오류를 범하지 않을수 있기 때문이다.

출처 : SQL Server Mag. 2003.11

PS. 다음은 차기버전부터는 T-SQL JOIN을 지원하지 않는다는 SQL 2000 북스 온라인 내용입니다.

**Transact-SQL 조인**

Microsoft® SQL Server™ 2000 이전 버전에서는 \*= 및 =\* 연산자를 사용하여 WHERE 절에 왼쪽 및 오른쪽 외부 조인 조건을 지정하였습니다. 경우에 따라 이러한 구문은 한 가지 방법 이상으로 해석될 수 있는 모호한 쿼리가 됩니다. SQL-92 규격 외부 조인은 FROM 절에 지정되어 이러한 모호성을 없애 줍니다. SQL-92 구문이 더 정확하기 때문에, WHERE 절에 이전 Transact-SQL 외부 조인 구문을 사용하는 것에 대한 자세한 내용은 이 릴리스에 포함되어 있지 않습니다. 앞으로 출시될 SQL Server 버전에서는 이러한 이전 구문을 지원하지 않을 것입니다. Transact-SQL 외부 조인을 사용하는 모든 명령은 SQL-92 구문을 사용하도록 변경해야 합니다.

SQL-92 표준은 FROM 또는 WHERE 절에 내부 조인을 지정하는 것을 지원하지 않습니다. WHERE 절에 지정된 내부 조인은 Transact-SQL 외부 조인 구문이 갖는 모호성 문제가 없습니다.