SQL Server – Joins

Stephan Karrer

Joins unter SQL Server

Syntax
`FROM first_table join_type second_table [ON (join_condition)]` *(ANSI-Syntax 99)*
`FROM first_table, second_table  WHERE join_condition` *(ANSI-Syntax 92)*

Unterstützte Arten:

- Inner Join (Equi Join als Spezialform)
- Self Join
- Cross Join (Kartesisches Produkt)
- Outer Join

Equi-Join (Spezialform des Inner-Join)

```
SELECT  employees.employee_id, employees.last_name,
employees.department_id, departments.location_id

        FROM employees INNER JOIN departments
        ON (employees.department_id = departments.department_id);


SELECT  e.employee_id, e.last_name, e.department_id,
        d.location_id

        FROM employees e JOIN departments d
        ON (e.department_id = d.department_id);
```

Equi-Join über WHERE-KLausel: alte Schreibweise

```
SELECT  e.employee_id, e.last_name, e.department_id,
        d.location_id

        FROM employees e, departments d

        WHERE e.department_id = d.department_id

                AND d.location_id > 1000;
```

Mehrfach-Join

```
/* Das Schlüsselwort "ON" muss nach dem jeweiligen "JOIN"
folgen  */


SELECT e.last_name, d.department_name, l.city, c.country_name
  FROM employees e INNER JOIN departments d
                     ON e.department_id = d.department_id
                   INNER JOIN locations l
                     ON d.location_id = l.location_id
                   INNER JOIN countries c
                     ON l.country_id = c.country_id ;
```

Non-Equi Join

```
select e.last_name, e.department_id,

       d.department_id, d.department_name

   from employees e JOIN departments d

         ON e.department_id > d.department_id

        WHERE e.department_id = 50 and e.employee_id = 140;
```

Outer Joins

```
SELECT  e.employee_id, e.last_name, e.department_id,
        d.department_id, d.department_name

        FROM employees e LEFT OUTER JOIN departments d
        ON (e.department_id = d.department_id);

SELECT d.department_id, e.last_name      -- Oracle-spez. Syntax
       FROM departments d, employees e
       WHERE d.department_id = e.department_id(+)
       ORDER BY d.department_id, e.last_name;

SELECT  e.employee_id, e.last_name, d.department_id,
        d.department_name

        FROM employees e FULL OUTER JOIN departments d
        ON (e.department_id = d.department_id);
```

Cross Join (Kartesisches Produkt)

```
SELECT  e.employee_id, e.last_name, e.department_id,
        d.department_id, d.department_name
        FROM employees e CROSS JOIN departments d
        WHERE e.department_id = d.department_id
        ORDER BY e.employee_id ;
```

Self Join

```
SELECT  e.last_name AS emp, m.last_name AS man
        FROM employees e INNER JOIN employees m
        ON (e.manager_id = m.employee_id) ;
```