# Tipps & Tricks: Gruppenfunktionen

| Bereich: | DBA, SQL | Erstellung: | 03/2004 HA |
|---|---|---|---|
| Versionsinfo: | 10.1, 10.2, 11.1, 11.2 | Letzte Überarbeitung: | 06/2009 MA |

## Gruppenfunktionen

Gruppenfunktionen sind SQL-Funktionen, die mehrere Zeilen einer Tabelle zu Gruppen zusammenfassen und für jede einzelne Gruppe ein Ergebnis zurückgeben. Diese Gruppen können die gesamte Tabelle oder verschiedene Tabellenbereiche umfassen.

Um eine Tabelle in kleinere Gruppen zu unterteilen, wird in der **GROUP BY-Klausel** angegeben, nach welcher Spalte (bzw. welchen Spalten) gruppiert werden soll (z.B. nach Abteilungsnummern).

Typen von Gruppenfunktionen

Als Gruppenfunktionen werden bei Oracle folgende Typen verwendet:

- AVG  Ermittelt den Durchschnittswert einer selektierten Rückgabemenge.
- COUNT  Gibt die Anzahl aller selektierten Zeilen zurück.
- MAX / MIN  Ermittelt den größten / kleinsten Wert einer Spalte oder Gruppe.
- STDDEV  Liefert die Standardabweichung aller Werte einer Spalte oder Gruppe.
- SUM  Ermöglicht das Aufsummieren von Werten einer Spalte oder Gruppe.
- VARIANCE  Liefert die Varianz von Werten einer Spalte oder Gruppe.

**Allgemeine Syntax:**

```
SQL> SELECT [spalte(n),] gruppen_funktion(spalte)
     FROM tabelle
     [WHERE bedingung]
     [GROUP BY spalte(n)]
     [HAVING bedingung]
     [ORDER BY spalte(n)];
```

**Hinweis:**

Alle Nichtgruppenfunktionsspalten aus der SELECT-Klausel müssen in der GROUP BY-Klausel aufgeführt werden, ansonsten erhält man die Fehlermeldung `ORA-00937: keine Gruppenfunktion für Einzelgruppe.` In der WHERE-Klausel können keine Gruppenfunktionen als Einschränkung angegeben werden. Sollen Gruppenergebnisse gefiltert werden, erfolgt dies über die HAVING-Klausel.

**Beispiele zu Gruppenfunktionen:**

1) Es sollen das Maximal-, das Minimal- und das Durchschnittsgehalt der EMP-Tabelle ausgegeben werden:

```
SQL> SELECT COUNT(*), MAX(sal), MIN(sal), AVG(sal), SUM(sal) FROM emp;
```

```
  COUNT(*)   MAX(SAL)   MIN(SAL)   AVG(SAL)   SUM(SAL)
---------- ---------- ---------- ---------- ----------
        14       5000        800 2073,21429      29025
```

2) Es sollen alle Datenbank-Benutzer und die Anzahl ihrer Objekte ermittelt werden.
   Hinweis: Für dieses Beispiel sind DBA Rechte erforderlich.

```
SQL> SELECT owner, COUNT(object_name) FROM dba_objects
     GROUP BY owner;
```

```
OWNER                            COUNT(*)
-------------------------------- ----------
CTXSYS                                  263
MDSYS                                   245
OEM_ORCL                                940
OEM_TAHOE_ORCL                          670
PUBLIC                                11541
RMAN                                    128
SCOTT                                    24
SYS                                   13528
SYSTEM                                  436
...
```

3) Für jede Berufsgruppe in den einzelnen Abteilungen soll das Gesamtgehalt ermittelt werden:

```
SQL> SELECT deptno, job, SUM(sal)
     FROM emp
     GROUP BY deptno, job;
```

```
   DEPTNO JOB          SUM(SAL)
---------- --------- ----------
        10 CLERK           1300
        10 MANAGER         2450
        10 PRESIDENT       5000
        20 CLERK           1900
        20 ANALYST         6000
        20 MANAGER         2975
        30 CLERK            950
        30 MANAGER         2850
        30 SALESMAN        5600
```

4) Gesucht sind alle Berufsgruppen einer Abteilung, in denen genau ein Mitarbeiter arbeitet und deren Durchschnittsgehalt über 2000 liegt.

```
SQL> SELECT deptno, job, COUNT(*), AVG(sal)
     FROM emp
     GROUP BY deptno, job
```

```
    HAVING COUNT(*) = 1 AND
    AVG(sal)>2000
```

```
DEPTNO     JOB         COUNT(*)   AVG(SAL)
---------- --------- ---------- ----------
        10 MANAGER            1       2450
        10 PRESIDENT          1       5000
        20 MANAGER            1       2975
        30 MANAGER            1       2850
```