

Vertiefung SQL

Data Query language

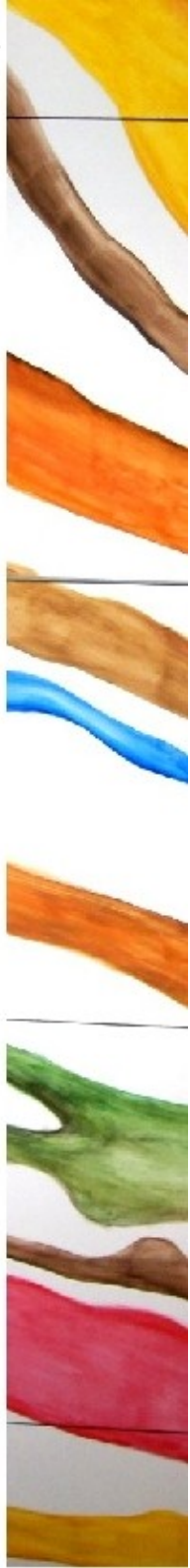


```
SELECT fahrraeder.Fahrradnr, fahrraeder.Bezeichnung,
fahrraeder.Tagesmietpreis
FROM fahrraeder
WHERE YEAR(fahrraeder.Kaufdatum) < 2009;
```

Autor: Christine Janischek
Thema: Vertiefung SQL

Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen



Wiederholung

Die FROM - Klausel	Hinter FROM steht der Name der Tabelle. Entstammen die Felder verschiedenen Tabellen, so muss jeweils der Tabellename durch einen Punkt getrennt vorangesetzt werden (z.B.: Kunden.Ortnr).
Die WHERE - Klausel	<p>Damit wird bestimmt, welche Datensätze (Zeilen einer Tabelle) ausgewählt werden sollen. (Bedingungsprüfung). Enthält die Bedingungsprüfung mehrere Bedingungen, werden diese mit logischen Operatoren AND, OR und NOT verknüpft:</p> <p>AND ist dann wahr, wenn beide Bedingungen/Kriterien erfüllt sind Beispiel: Alle weiblichen Kunden aus Filderstadt (Ortnr=29740). ... Geschl = "w" AND Ortnr = 29740</p> <p>OR ist dann wahr, wenn mindestens eine der beiden Bedingungen/Kriterien erfüllt ist. Beispiel: Alle Kunden aus Blaubeuren (Ortnr 39815) und Laichingen (Ortnr 39816). ... OrtNr = 39815 OR Ortnr = 39816</p> <p>NOT negiert einen Ausdruck. Beispiel: Alle Kunden, die nicht aus Stuttgart kommen. ... NOT Ort = "Stuttgart"</p>

Autor: Christine Janischek
Thema: Vertiefung SQL

Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen

Neue Klauseln

Hinweis: MySQL verwendet als Platzhalter für beliebige Zeichen innerhalb einer Zeichenkette das "%" -Symbol. In Access findet das "*" -Symbol Anwendung.

Die ORDER BY - Klausel	Daten werden nach einem oder mehreren Feldnamen sortiert ausgegeben. Die vorgegebene Sortierreihenfolge ist aufsteigend ASC(ending) . Soll absteigend sortiert werden, muss DESC(ending) eingegeben werden (...ORDER BY wert DESC).
Die ORDER BY - Klausel	Daten werden nach einem oder mehreren Feldnamen sortiert ausgegeben. Die vorgegebene Sortierreihenfolge ist aufsteigend ASC(ending) . Soll absteigend sortiert werden, muss DESC(ending) eingegeben werden (...ORDER BY wert DESC).
Die GROUP BY - Klausel	Sie dient dazu, die Zeilen einer Tabelle nach bestimmten Feldern zu gruppieren.
Die HAVING - Klausel	Bei der HAVING -Klausel werden bestimmte Gruppen aus einer Gruppenmenge ausgewählt. Somit kann die HAVING -Klausel nur im Zusammenhang mit der GROUP BY -Klausel angewandt werden.

Beispiel - Fragestellung:

Alle Fahrräder, deren Bezeichnung mit ‚Scott‘ beginnt, soll aufgelistet werden. Die Liste sollen nach dem Kaufdatum absteigend sortiert sein.

Weitere Vergleichsoperatoren

SQL kennt die üblichen Vergleichsoperatoren:

= gleich <> ungleich > größer < kleiner >= größer gleich <= kleiner gleich

BETWEEN <i>.Wert1..AND..Wert2..</i>	Vergleichswert liegt <i>zwischen</i> Wert1 und Wert2
IN <i>Werteliste</i>	Vergleichswert ist in der angegebenen <i>Werteliste</i>
Like <i>Zeichenfolge</i>	Vergleichszeichen entsprechen der <i>Zeichenfolge</i>
Is Null <i>Feld</i>	Vergleichsfeld hat einen NULL-Wert
Vergleichsoperatoren können verknüpft werden mit den Operatoren AND , OR und NOT	

Rechenoperationen (+ , - , * , /) sind nur möglich mit numerischen Daten.

Beispiel – Fragestellung:

Aufzulisten sind alle Fahrräder, deren Tagesmietpreis zwischen 15,00 € und 20,00 € liegt (Attribute: *Fahrradnr*, *Bezeichnung* und *Tagesmietpreis*).

Gruppierungsfunktionen (Aggregatsfunktionen)

AVG (Spalte)	= Durchschnittswert	COUNT (Spalte)	= Anzahl aller Einträge
MAX (Spalte)	= Maximalwert	MIN (Spalte)	= Minimalwert
SUM (Spalte)	= Summe aller Einträge in einer Spalte		

Gruppierungsfunktionen können nur anstelle eines Spaltennamens direkt hinter der SELECT-Anweisung stehen. Sie liefern genau einen Wert, beziehen sich jedoch auf mehrere Tabellenzeilen.

Beispiel – Fragestellung:

Ermitteln Sie den Gesamtwert aller Fahrräder, deren durchschnittlichen Wert sowie den Wert des teuersten und des günstigsten Fahrrades.

Autor: Christine Janischek
Thema: Vertiefung SQL

Kaufmännische
Schule
Wangen

Wege zeigen, öffnen, gehen