

Einfacher geht's nicht!

SELECT * FROM LND;

LANDSCHL	BEZEICHN
BM	Bermudas
DK	
E	
F	
GB	
GCA	Guatemala
I	
NL	
PNG	Papua Neuguinea
USA	

SELECT * FROM KND;

KUNDNR	KURZNAME	PKTOFIBU	LAND	KRELIMIT	PRIVAT	ANLDAT
1	ABZOCKER, BERLIN	10001		1000000,00	N	27.04.2006
2	MADGIRLS, LONDON	10002	GB		N	27.04.2006
3	STUPIDO, ROMA	10003	I	2,50	N	29.04.2006
4	FRAUDEUR, PARIS	10004	F	3000,00	N	29.04.2006
5	BRITNEY SPEARS, LA	10005	USA	1,00	J	30.04.2006
6	BRAD PITT, NY	10006	USA	1,50	J	30.04.2006

SELECT * FROM LIE;

LIEFNR	KURZNAME	PKTOFIBU	LAND	KRELIMIT	ANLDAT
1	WEBER, PADERBORN	70001		10000,00	27.04.2006
2	MILLER, HAMILTON	70002	BM		27.04.2006
3	LAHO, BAGNEUX	70012	F	5000,00	28.04.2006
4	FOSTERS, BIRMINGHAM	70004	GB	7500,00	30.04.2006

SELECT * FROM MEH;

MENGEINH
Kg
Lit
Std
Stk

SELECT * FROM ART;

ARTIKEL	BEZEICHN	HPTLIE	MEH	PREIS1	PRS2AB	PREIS2
10.1001	Gartenpumpe	1	Stk	39,99	01.06.2006	49,99
10.1016	Hochdruckreiniger	1	Stk	99,99		
25.3282	Holzlasur	2	Lit	4,98	01.04.2006	3,98
56.7954	Fliesenkleber	3	Kg	0,40		
80.0001	Montageleistungen		Std	49,95		

Spaltenauswahl

In den wenigsten Fällen ist die Ausgabe sämtlicher Spalten erforderlich. Deshalb ist es sinnvoll, nur die tatsächlich benötigten Spalten auszuwählen, um den Umfang der Ergebnismenge auf das notwendige Maß zu beschränken. Dieses Vorgehen führt zu einer Reduzierung des Datentransfers und bewirkt eine Steigerung der Systemleistung.

Das Symbol * wird durch eine Aufzählung der gewünschten Spaltennamen ersetzt. Die einzelnen Spaltennamen werden dabei durch Komma getrennt.

Beispiele

```
SELECT LANDSCHL FROM LND;
```

LANDSCHL
BM
DK
E
F
GB
GCA
I
NL
PNG
USA

```
SELECT LIEFNR, KURZNAME FROM LIE;
```

LIEFNR	KURZNAME
1	WEBER, PADERBORN
2	MILLER, HAMILTON
3	LAHO, BAGNEUX
4	FOSTERS, BIRMINGHAM

```
SELECT ARTIKEL, BEZEICHN, MEH FROM ART;
```

ARTIKEL	BEZEICHN	MEH
10.1001	Gartenpumpe	Stk
10.1016	Hochdruckreiniger	Stk
25.3282	Holzlasur	Lit
56.7954	Fliesenkleber	Kg
80.0001	Montageleistungen	Std

Bedingungen

Bedingungen werden mit der Klausel **WHERE** formuliert. Diese Klausel haben wir bereits im Zusammenhang mit UPDATE- bzw. DELETE-Anweisungen genutzt, um eine Änderungs- bzw. Löschk Aktion auf genau einen spezifizierten Primärschlüsselinhalt zu beschränken.

Merke: Ohne Angabe einer WHERE-Bedingung werden alle Sätze einer Tabelle in eine Aktion einbezogen.

Vergleichsoperatoren

- = gleich
- < kleiner
- <= kleiner oder gleich
- > größer
- >= größer oder gleich
- <> ungleich

Vergleichsverfahren

Felder vom Typ NUMBER bzw. DECIMAL und Felder vom Typ DATE bzw. TIMESTAMP werden aufgrund ihres Wertes verglichen.

Bei Feldern des Typs VARCHAR bzw. VARCHAR2 werden die einzelnen Zeichen stellengerecht mit der Wertigkeit gemäß Zeichensatz verglichen. Felder dieses Typs werden wie Zeichenketten in C behandelt und sind durch ein unsichtbares Zeichen mit der Wertigkeit Null terminiert. Der Vergleichsprozess endet mit der ersten Ungleichheit.

Internationaler Windows-Zeichensatz (WIN1252) in hexadezimaler Wertigkeit

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				0	@	P	`	p	€	□		°	À	Đ	à	Đ
1			!	1	A	Q	a	q	□	'	ı	±	Á	Ñ	á	ñ
2			"	2	B	R	b	r	,	'	ç	²	Â	Ò	â	ò
3			#	3	C	S	c	s	f	“	£	³	Ã	Ó	ã	ó
4			\$	4	D	T	d	t	„	”	¤	´	Ä	Ô	ä	ô
5			%	5	E	U	e	u	...	•	¥	µ	Å	Õ	å	õ
6			&	6	F	V	f	v	†	-	ı	¶	Æ	Ö	æ	ö
7			'	7	G	W	g	w	‡	—	§	·	Ç	×	ç	÷
8			(8	H	X	h	x	^	~	¨	¸	È	Ø	è	ø
9)	9	I	Y	i	y	‰	™	©	¹	É	Ù	é	ù
A			*	:	J	Z	j	z	Š	š	ª	º	Ê	Ú	ê	ú
B			+	;	K	[k	{	<	>	«	»	Ë	Û	ë	û
C			,	<	L	\	l		Œ	œ	¬	¼	Ì	Ü	ì	ü
D			-	=	M]	m	}	□	□	-	½	Í	Ý	í	ý
E			.	>	N	^	n	~	Ž	ž	®	¾	Î	Þ	î	þ
F			/	?	O	_	o	□	□	ÿ	-	¿	Ï	ß	ï	ÿ

Vergleichsbeispiele:

'DU' im Vergleich zu 'Du'

Zeichen	D	U	
Hexadezimal	44	55	0
Dezimal	68	85	0

Zeichen	D	u	
Hexadezimal	44	75	0
Dezimal	68	117	0

Das erste Zeichen ist in beiden Zeichenketten gleich. Das zweite Zeichen 'u' der Zeichenkette 'Du' ist größer als das zweite Zeichen 'U' der Zeichenkette 'DU'. Der Vergleich beider Zeichenketten wird beendet und ergibt:

'DU' ist kleiner als 'Du'

'Anna' im Vergleich zu 'Ann'

Zeichen	A	n	n	a	
Hexadezimal	41	6E	6E	61	0
Dezimal	65	110	110	97	0

Zeichen	A	n	n	
Hexadezimal	41	6E	6E	0
Dezimal	65	110	110	0

Die ersten drei Zeichen sind in beiden Zeichenketten identisch. Das 4. Zeichen der Zeichenkette 'Anna' hat die dezimale Wertigkeit 97 und ist damit größer als das 4. und letzte Zeichen der Zeichenkette 'Ann' mit der dezimalen Wertigkeit Null. Der Vergleich wird an dieser Stelle beendet und ergibt:

'Ann' ist kleiner als 'Anna'

Anwendungsbeispiele:

```
SELECT BEZEICHN FROM ART WHERE ARTIKEL='10.1016';
```

BEZEICHN
Hochdruckreiniger

```
SELECT KUNDNR, KURZNAME FROM KND WHERE KUNDNR < 5;
```

KUNDNR	KURZNAME
1	ABZOCKER, BERLIN
2	MADGIRLS, LONDON
3	STUPIDO, ROMA
4	FRAUDEUR, PARIS

```
SELECT MENGEINH FROM MEH WHERE MENGEINH <= 'Std';
```

MENGEINH
Kg
Lit
Std

```
SELECT LIEFNR, KURZNAME FROM LIE WHERE KURZNAME > 'FOSTERS';
```

LIEFNR	KURZNAME
1	WEBER, PADERBORN
2	MILLER, HAMILTON
3	LAHO, BAGNEUX
4	FOSTERS, BIRMINGHAM

Anmerkung: 'FOSTERS, BIRMINGHAM' ist größer als 'FOSTERS'

```
SELECT LANDSCHL FROM LND WHERE LANDSCHL >= 'usa';
```

LANDSCHL

Anmerkung: 'u' ist größer als 'U', deshalb keine Selektion

Oracle:

```
SELECT LIEFNR, KURZNAME FROM LIE WHERE ANLDAT >= TO_DATE('28.04.2006', 'DD.MM.YYYY');
```

Interbase:

```
SELECT LIEFNR, KURZNAME FROM LIE WHERE ANLDAT >= '28-APR-2006';
```

LIEFNR	KURZNAME
3	LAHO, BAGNEUX
4	FOSTERS, BIRMINGHAM

```
SELECT ARTIKEL, BEZEICHN FROM ART WHERE PREIS2 <> 0;
```

ARTIKEL	BEZEICHN
10.1001	Gartenpumpe
25.3282	Holzlasur

Logische Operatoren

Mehrere Bedingungen können mit den Operatoren

- AND
- OR
- NOT

verknüpft werden.

Die Reihenfolge der Auswertung von Ausdrücken kann durch Klammerung beeinflusst werden. Im Übrigen gelten die Regeln der Aussagenlogik (Boolsche Algebra) in vollem Umfang.

Beispiele

```
SELECT KUNDNR, KURZNAME, PKTOFIBU FROM KND
WHERE PKTOFIBU >= 10002 AND PKTOFIBU <= 10004 OR PKTOFIBU = 10006;
```

KUNDNR	KURZNAME	PKTOFIBU
2	MADGIRLS, LONDON	10002
3	STUPIDO, ROMA	10003
4	FRAUDEUR, PARIS	10004
6	BRAD PITT, NY	10006

```
SELECT KUNDNR, KURZNAME, PKTOFIBU FROM KND
WHERE NOT (PKTOFIBU >= 10002 AND PKTOFIBU <= 10004 OR PKTOFIBU = 10006);
```

KUNDNR	KURZNAME	PKTOFIBU
1	ABZOCKER, BERLIN	10001
5	BRITNEY SPEARS, LA	10005

```
SELECT KUNDNR, KURZNAME, PKTOFIBU FROM KND
WHERE NOT PKTOFIBU >= 10002 AND PKTOFIBU <= 10004 OR PKTOFIBU = 10006;
```

KUNDNR	KURZNAME	PKTOFIBU
1	ABZOCKER, BERLIN	10001
6	BRAD PITT, NY	10006

```
SELECT KUNDNR, KURZNAME, PKTOFIBU, PRIVAT FROM KND
WHERE PKTOFIBU >= 10002 OR PKTOFIBU <= 10004 OR PKTOFIBU = 10006 AND PRIVAT = 'J';
```

KUNDNR	KURZNAME	PKTOFIBU	PRIVAT
1	ABZOCKER, BERLIN	10001	N
2	MADGIRLS, LONDON	10002	N
3	STUPIDO, ROMA	10003	N
4	FRAUDEUR, PARIS	10004	N
5	BRITNEY SPEARS, LA	10005	J
6	BRAD PITT, NY	10006	J

```
SELECT KUNDNR, KURZNAME, PKTOFIBU, PRIVAT FROM KND
WHERE (PKTOFIBU >= 10002 OR PKTOFIBU <= 10004 OR PKTOFIBU = 10006) AND PRIVAT = 'J';
```

KUNDNR	KURZNAME	PKTOFIBU	PRIVAT
5	BRITNEY SPEARS, LA	10005	J
6	BRAD PITT, NY	10006	J

NULL-Bedingungen

Die regulären Vergleichsoperatoren dürfen auf Bedingungen bezüglich der Konstanten NULL nicht eingesetzt werden. Für diese Fälle sind zwei Sonderkonstruktionen (IS NULL) bzw. (IS NOT NULL) vorgesehen.

Beispiele

```
SELECT KUNDNR, KURZNAME FROM KND WHERE LAND IS NULL ;
```

KUNDNR	KURZNAME
1	ABZOCKER, BERLIN

```
SELECT LANDSCHL, BEZEICHN FROM LND WHERE BEZEICHN IS NOT NULL ;
```

LANDSCHL	BEZEICHN
BM	Bermudas
GCA	Guatemala
PNG	Papua Neuguinea

BETWEEN minvalue AND maxvalue

Mit der Klausel **BETWEEN** kann ein Wertebereich als Bedingung formuliert werden.

Eine Sql-Abfrage zur Selektion aller Kunden mit einem zugeordneten Kreditlimit zwischen 1000 € und 9999999 € könnte z.B. die nachfolgende Bedingung enthalten:

```
SELECT KUNDNR, KURZNAME, KRELIMIT FROM KND WHERE KRELIMIT BETWEEN 1000 AND 9999999 ;
```

KUNDNR	KURZNAME	KRELIMIT
1	ABZOCKER, BERLIN	1000000,00
4	FRAUDEUR, PARIS	3000,00

IN (coldata[, coldata...])

Die Klausel **IN** kann nicht nur in CHECK-Bedingungen, sondern auch in SQL-Abfragen genutzt werden.

Die Selektion aller Teile mit den Mengeneinheiten 'Stk' oder 'Kg' kann z.B. wie folgt formuliert werden:

```
SELECT ARTIKEL, BEZEICHN, MEH FROM ART WHERE MEH IN ('Stk', 'Kg');
```

ARTIKEL	BEZEICHN	MEH
10.1001	Gartenpumpe	Stk
10.1016	Hochdruckreiniger	Stk
56.7954	Fliesenkleber	Kg

UPPER(colname)

Kleinbuchstaben in Spalteninhalten können mit der Funktion **UPPER** in Großbuchstaben konvertiert werden.

```
SELECT ARTIKEL, UPPER(BEZEICHN) FROM ART;
```

ARTIKEL	UPPER(BEZEICHN)
10.1001	GARTENPUMPE
10.1016	HOCHDRUCKREINIGER
25.3282	HOLZLASUR
56.7954	FLIESENKLEBER
80.0001	MONTAGELEISTUNGEN

Abschneiden der Uhrzeit

Wie bei den Check-Bedingungen kann in Oracle die Funktion **TRUNC** bzw. in Interbase die Funktion **CAST** genutzt werden, um bei Feldern des Typs DATE bzw. TIMESTAMP den Uhrzeitanteil bei der Anzeige zu unterdrücken.

Oracle: SELECT LIEFNR, KURZNAME, TRUNC(ANLDAT) FROM LIE;

LIEFNR	KURZNAME	TRUNC(ANLDAT)
1	WEBER, PADERBORN	27.04.2006
2	MILLER, HAMILTON	27.04.2006
3	LAHO, BAGNEUX	28.04.2006
4	FOSTERS, BIRMINGHAM	30.04.2006

Interbase: SELECT LIEFNR, KURZNAME, CAST(ANLDAT AS DATE) FROM LIE;

LIKE mask

Felder des Typs VARCHAR bzw. VARCHAR2 können mit maskierten Teilzeichenketten verglichen werden.

Die Vergleichsmaske kann mit den Wildcardsymbolen '%' und '_' definiert werden. Dieser Mechanismus ist vergleichbar mit der Suche nach Dateien bzw. Ordnern in Windows. In SQL ist das Zeichen '%' identisch mit dem Zeichen '*' und das Zeichen '_' identisch mit dem Zeichen '?' in Windows.

Wildcards:

% Alle Zeichen vor bzw. nach diesem Zeichen sind irrelevant

_ Der Inhalt dieser Stelle innerhalb der Zeichenkette ist irrelevant

Beispiele

SELECT LANDSCHL, BEZEICHN FROM LND WHERE LANDSCHL LIKE 'G_';

LANDSCHL	BEZEICHN
GB	

SELECT LANDSCHL, BEZEICHN FROM LND WHERE LANDSCHL LIKE 'G_%';

LANDSCHL	BEZEICHN
GB	
GCA	Guatemala

SELECT LANDSCHL, BEZEICHN FROM LND WHERE LANDSCHL LIKE '%G%';

LANDSCHL	BEZEICHN
GB	
GCA	Guatemala
PNG	Papua Neuguinea

SELECT ARTIKEL, BEZEICHN FROM ART WHERE UPPER(BEZEICHN) LIKE '%EN%';

ARTIKEL	BEZEICHN
10.1001	Gartenpumpe
56.7954	Fliesenkleber
80.0001	Montageleistungen

SELECT ARTIKEL, BEZEICHN FROM ART WHERE UPPER(BEZEICHN) LIKE 'EN';

ARTIKEL	BEZEICHN
80.0001	Montageleistungen

Sortierung

Die Reihenfolge der Sätze in der Ergebnismenge kann durch die Klausel **ORDER BY** festgelegt werden. Das Festlegen einer bestimmten Sortierreihenfolge bewirkt einen erheblichen Mehraufwand beim Zusammenstellen der Ergebnismenge durch das Datenbankmanagementsystem. Eine Sortierfolge sollte daher nur dann festgelegt werden, wenn diese tatsächlich relevant ist.

Syntax: ORDER BY colname [DESC] [, colname [DESC] ,...]

Standardmäßig werden die Daten einer Spalte aufsteigend sortiert. Durch Angabe des Schlüsselworts **DESC** kann eine absteigende Sortierfolge bestimmt werden.

Alternativ zu den Spaltennamen kann auch die Position der Spalte in der Ergebnismenge (von links nach rechts beginnend mit 1) als Sortierkriterium angegeben werden. Aus Gründen der Klarheit und Übersichtlichkeit sollte diese Möglichkeit allerdings nur in Ausnahmefällen, z.B. bei berechneten Spalteninhalten, benutzt werden. Berechnete Spalteninhalte werden zu einem späteren Zeitpunkt behandelt.

Anwendungsbeispiele:

```
SELECT KUNDNR, KURZNAME, LAND FROM KND ORDER BY KUNDNR;
```

KUNDNR	KURZNAME	LAND
1	ABZOCKER, BERLIN	
2	MADGIRLS, LONDON	GB
3	STUPIDO, ROMA	I
4	FRAUDEUR, PARIS	F
5	BRITNEY SPEARS, LA	USA
6	BRAD PITT, NY	USA

```
SELECT KUNDNR, KURZNAME, LAND FROM KND ORDER BY 1 DESC;
```

KUNDNR	KURZNAME	LAND
6	BRAD PITT, NY	USA
5	BRITNEY SPEARS, LA	USA
4	FRAUDEUR, PARIS	F
3	STUPIDO, ROMA	I
2	MADGIRLS, LONDON	GB
1	ABZOCKER, BERLIN	

```
SELECT KUNDNR, KURZNAME, LAND FROM KND WHERE KUNDNR > 1 ORDER BY LAND, KUNDNR DESC;
```

KUNDNR	KURZNAME	LAND
4	FRAUDEUR, PARIS	F
2	MADGIRLS, LONDON	GB
3	STUPIDO, ROMA	I
6	BRAD PITT, NY	USA
5	BRITNEY SPEARS, LA	USA

```
SELECT KUNDNR, KURZNAME, LAND FROM KND WHERE KUNDNR > 1 ORDER BY LAND, KUNDNR;
```

KUNDNR	KURZNAME	LAND
4	FRAUDEUR, PARIS	F
2	MADGIRLS, LONDON	GB
3	STUPIDO, ROMA	I
5	BRITNEY SPEARS, LA	USA
6	BRAD PITT, NY	USA

Übungsaufgaben

1. Selektieren Sie die Sätze aus Tabelle ART bei denen der Wert der Spalte PREIS1 größer oder gleich 10 € ist und bei denen die Spalte PRS2AB leer ist. Die Ausgabe soll die Spalten ARTIKEL, BEZEICHN und PREIS1 umfassen. Das Ergebnis soll absteigend nach der Spalte ARTIKEL sortiert sein.

Ausgabebeispiel:

ARTIKEL	BEZEICHN	PREIS1
80.0001	Montageleistungen	49,95
10.1016	Hochdruckreiniger	99,99

2. Selektieren Sie die Lieferanten aus Tabelle LIE, denen kein Kreditlimit zugeordnet ist oder die im Ausland ansässig sind. Die Ausgabe soll die Spalten LIEFNR und KURZNAME umfassen. Das Ergebnis soll aufsteigend nach Lieferantenummer (LIEFNR) sortiert sein.

Ausgabebeispiel:

LIEFNR	KURZNAME
2	MILLER, HAMILTON
3	LAHO, BAGNEUX
4	FOSTERS, BIRMINGHAM

3. Selektieren Sie alle Geschäftskunden (PRIVAT='N') aus Tabelle KND, denen ein Kreditlimit zugeordnet ist. Die Ergebnismenge soll die Spalten KUNDNR, KURZNAME und KRELIMIT umfassen und aufsteigend nach Kreditlimit sortiert sein.

Ausgabebeispiel:

KUNDNR	KURZNAME	KRELIMIT
3	STUPIDO, ROMA	2,50
4	FRAUDEUR, PARIS	3000,00
1	ABZOCKER, BERLIN	1000000,00

4. Selektieren Sie alle Lieferanten mit den Spalten LIEFNR und KURZNAME, in deren Kurznamen die Zeichenkette 'ER' an beliebiger Stelle enthalten ist.

Ausgabebeispiel:

LIEFNR	KURZNAME
1	WEBER, PADERBORN
2	MILLER, HAMILTON
4	FOSTERS, BIRMINGHAM

5. Selektieren Sie alle Artikel mit den Spalten ARTIKEL und BEZEICHN, in deren Artikelnummer das zweite und das fünfte Zeichen '0' ist. Die Sortierung soll absteigend nach Artikelnummer erfolgen.

Ausgabebeispiel:

ARTIKEL	BEZEICHN
80.0001	Montageleistungen
10.1016	Hochdruckreiniger
10.1001	Gartenpumpe

Lösungen

1.

```
SELECT ARTIKEL, BEZEICHN, PREIS1 FROM ART
WHERE PREIS1 >= 10 AND PRS2AB IS NULL
ORDER BY ARTIKEL DESC;
```
2.

```
SELECT LIEFNR, KURZNAME FROM LIE
WHERE KRELIMIT=0 OR LAND IS NOT NULL
ORDER BY LIEFNR;
```
3.

```
SELECT KUNDNR, KURZNAME, KRELIMIT FROM KND
WHERE PRIVAT='N' AND KRELIMIT <> 0
ORDER BY KRELIMIT;
```
4.

```
SELECT LIEFNR, KURZNAME FROM LIE
WHERE KURZNAME LIKE '%ER%';
```
5.

```
SELECT ARTIKEL, BEZEICHN FROM ART
WHERE ARTIKEL LIKE '_0__0%'
ORDER BY ARTIKEL DESC;
```

Daten aus mehreren Tabellen in einer Ergebnismenge

Daten aus unterschiedlichen Tabellen können in einer Ergebnismenge zusammengefasst werden. Die benötigten Tabellen werden in der **FROM**-Klausel angegeben. Bedingungen zur Bildung einer sinnvollen Schnittmenge werden in der **WHERE**-Klausel formuliert. Identische Spaltennamen in mehreren beteiligten Tabellen können durch Voranstellen des Tabellennamens eindeutig spezifiziert werden.

Im nachfolgenden Beispiel werden alle Teile aus der Tabelle ART, denen ein Hauptlieferant zugeordnet ist, mit Artikelnummer, Bezeichnung, Lieferantenummer und dem Kurznamen des Lieferanten, aufsteigend sortiert nach Artikelnummer, ausgegeben.

```
SELECT ART.ARTIKEL, ART.BEZEICHN, ART.HPTLIE, LIE.KURZNAME FROM ART, LIE
WHERE LIE.LIEFNR=ART.HPTLIE ORDER BY ART.ARTIKEL;
```

ARTIKEL	BEZEICHN	HPTLIE	KURZNAME
10.1001	Gartenpumpe	1	WEBER, PADERBORN
10.1016	Hochdruckreiniger	1	WEBER, PADERBORN
25.3282	Holzlasur	2	MILLER, HAMILTON
56.7954	Fliesenkleber	3	LAHO, BAGNEUX

Anmerkungen

- In der From-Klausel sind die beteiligten Tabellen (ART, LIE) spezifiziert.
- Die Schnittmenge wird aufgrund der Bedingung (LIE.LIEFNR=ART.HPTLIE) gebildet.
- Die Spalten sind durch den Tabellennamen, das Trennzeichen '.' und den Spaltennamen eindeutig spezifiziert.

Hinweis

Der Satz mit der Artikelnummer '80.0001' wird unterdrückt, da die Schnittmenge mit der Lieferantentabelle, aufgrund der fehlenden Angabe des Hauptlieferanten, leer ist.

Grundsätzlich können aber auch Datensätze mit leeren Schnittmengenbeziehungen angezeigt werden. Die thematische Behandlung solcher Konstellationen erfolgt jedoch erst zu einem späteren Zeitpunkt.

Ohne Angabe einer Schnittmengenbedingung wird jeder Satz aus Tabelle ART mit jedem Satz aus Tabelle LIE kombiniert. Das Ergebnis ist ein kartesisches Produkt aus den Sätzen der Tabelle ART und LIE.

```
SELECT ARTIKEL, BEZEICHN, HPTLIE, KURZNAME FROM ART, LIE ORDER BY ARTIKEL;
```

ARTIKEL	BEZEICHN	HPTLIE	KURZNAME
10.1001	Gartenpumpe	1	WEBER, PADERBORN
10.1001	Gartenpumpe	1	MILLER, HAMILTON
10.1001	Gartenpumpe	1	FOSTERS, BIRMINGHAM
10.1001	Gartenpumpe	1	LAHO, BAGNEUX
10.1016	Hochdruckreiniger	1	WEBER, PADERBORN
10.1016	Hochdruckreiniger	1	FOSTERS, BIRMINGHAM
10.1016	Hochdruckreiniger	1	MILLER, HAMILTON
10.1016	Hochdruckreiniger	1	LAHO, BAGNEUX
25.3282	Holzlasur	2	WEBER, PADERBORN
25.3282	Holzlasur	2	LAHO, BAGNEUX
25.3282	Holzlasur	2	FOSTERS, BIRMINGHAM
25.3282	Holzlasur	2	MILLER, HAMILTON
56.7954	Fliesenkleber	3	WEBER, PADERBORN
56.7954	Fliesenkleber	3	MILLER, HAMILTON
56.7954	Fliesenkleber	3	FOSTERS, BIRMINGHAM
56.7954	Fliesenkleber	3	LAHO, BAGNEUX
80.0001	Montageleistungen		WEBER, PADERBORN
80.0001	Montageleistungen		MILLER, HAMILTON
80.0001	Montageleistungen		FOSTERS, BIRMINGHAM
80.0001	Montageleistungen		LAHO, BAGNEUX

Da in der Tabelle ART 5 Sätze und in der Tabelle LIE 4 Sätze gespeichert sind, enthält die Ergebnismenge genau 20 (5 x 4) Sätze.

Die Spezifizierung der Spaltennamen durch Voranstellen des Tabellennamens ist in beiden Beispielen nicht notwendig, da Konflikte durch mehrfach vorhandene Spaltennamen bei dieser Konstellation nicht auftreten.

Empfehlungen zu Abfragen mit mehreren Tabellen

- Zur Vermeidung von kartesischen Produkten ist die Angabe von Schnittmengenbedingungen immer erforderlich.
- Aus Gründen der Klarheit und Übersichtlichkeit sollte der Tabellename den Spaltennamen vorangestellt werden.
- Die Angabe des Synonyms * zur Selektion aller Spalten bezieht sich ausschließlich auf die erstgenannte Tabelle in der FROM-Klausel und ist daher nicht sinnvoll.

Übungsaufgabe

Selektieren Sie alle ausländischen Lieferanten mit den Spalten LIEFNR, KURZNAME und LAND sowie der Bezeichnung des Landes gemäß Tabelle LND. Sortieren Sie das Ergebnis nach der Spalte LAND.

Ausgabebeispiel:

LIEFNR	KURZNAME	LAND	BEZEICHN
2	MILLER, HAMILTON	BM	Bermudas
3	LAHO, BAGNEUX	F	
4	FOSTERS, BIRMINGHAM	GB	

Lösung

```
SELECT LIE.LIEFNR, LIE.KURZNAME, LIE.LAND, LND.BEZEICHN FROM LIE, LND
WHERE LND.LANDSCHL=LIE.LAND ORDER BY LIE.LAND;
```

Spezielle Optionen

Unterdrücken identischer Zeileninhalte

Durch Angabe der Schlüsselworts **DISTINCT** direkt im Anschluss an das Schlüsselwort SELECT können doppelte Zeileninhalte unterdrückt werden.

Beispiel

```
SELECT DISTINCT MEH FROM ART ORDER BY MEH;
```

MEH
Kg
Lit
Std
Stk

Umbenennen von Spalten bzw. Ausdrücken

Spalten bzw. Ausdrücke einer Ergebnismenge können mit einem anderen Namen versehen werden. Dies ist insbesondere beim Einsatz von Funktionen bzw. berechneten Ausdrücken sinnvoll. Die meisten Softwareentwicklungssysteme erwarten darüber hinaus Spaltennamen, die bestimmten Konventionen entsprechen müssen.

Syntax

```
expression AS colname
```

Beispiel Oracle

```
SELECT ARTIKEL, UPPER(BEZEICHN) AS BEZEICHN, PREIS1, TRUNC(PRS2AB) AS ABDAT, PREIS2
FROM ART ORDER BY ARTIKEL;
```

Beispiel Interbase

```
SELECT ARTIKEL, UPPER(BEZEICHN) AS BEZEICHN, PREIS1, CAST(PRS2AB AS DATE) AS ABDAT, PREIS2
FROM ART ORDER BY ARTIKEL;
```

ARTIKEL	BEZEICHN	PREIS1	ABDAT	PREIS2
10.1001	GARTENPUMPE	39,99	01.06.2006	49,99
10.1016	HOCHDRUCKREINIGER	99,99		0,00
25.3282	HOLZLASUR	4,98	01.04.2006	3,98
56.7954	FLIESENKLEBER	0,40		0,00
80.0001	MONTAGELEISTUNGEN	49,95		0,00

Berechnete Spalten

Spalten des Datentyps NUMBER bzw. DECIMAL oder DATE bzw. TIMESTAMP können zur Formulierung arithmetischer Ausdrücke benutzt werden.

Beispiele

```
SELECT ANLDAT AS ANLAGE,ANLDAT+0.5 AS MITTAG,ANLDAT+1 AS DANACH
FROM KND WHERE KUNDNR=1;
```

ANLAGE	MITTAG	DANACH
27.04.2006 00:00:00	27.04.2006 12:00:00	28.04.2006 00:00:00

```
SELECT ARTIKEL, BEZEICHN, PREIS1 AS NETTO, PREIS1*1.16 AS BRUTTO
FROM ART ORDER BY ARTIKEL;
```

ARTIKEL	BEZEICHN	NETTO	BRUTTO
10.1001	Gartenpumpe	39,99	46,3884
10.1016	Hochdruckreiniger	99,99	115,9884
25.3282	Holzlasur	4,98	5,7768
56.7954	Fliesenkleber	0,40	0,4640
80.0001	Montageleistungen	49,95	57,9420

Das Ergebnis berechneter Spalten kann auch gerundet werden.

Oracle (Funktion ROUND)

```
SELECT ARTIKEL, BEZEICHN, PREIS1 AS NETTO, ROUND(PREIS1*1.16,2) AS BRUTTO
FROM ART ORDER BY ARTIKEL;
```

Interbase (Funktion CAST)

```
SELECT ARTIKEL, BEZEICHN, PREIS1 AS NETTO, CAST(PREIS1*1.16 AS DECIMAL(8,2)) AS BRUTTO
FROM ART ORDER BY ARTIKEL;
```

ARTIKEL	BEZEICHN	NETTO	BRUTTO
10.1001	Gartenpumpe	39,99	46,39
10.1016	Hochdruckreiniger	99,99	115,99
25.3282	Holzlasur	4,98	5,78
56.7954	Fliesenkleber	0,40	0,46
80.0001	Montageleistungen	49,95	57,94

Konvertierung in VARCHAR bzw. VARCHAR2

Daten vom Typ NUMBER bzw. DECIMAL oder DATE bzw. TIMESTAMP können in Oracle mit der Funktion **TO_CHAR** bzw. in Interbase mit der Funktion **CAST** in den Typ VARCHAR2 bzw. VARCHAR konvertiert werden.

Beispiel (Oracle)

```
SELECT KUNDNR, KURZNAME, TO_CHAR(KRELIMIT) AS KRELIMIT,
TO_CHAR(ANLDAT, 'DD.MM.YYYY') AS ANLDAT FROM KND WHERE KUNDNR=1;
```

Beispiel (Interbase)

```
SELECT KUNDNR, KURZNAME, CAST(KRELIMIT AS VARCHAR(20)) AS KRELIMIT,
CAST(CAST(ANLDAT AS DATE) AS VARCHAR(20)) AS ANLDAT FROM KND WHERE KUNDNR=1;
```

KUNDNR	KURZNAME	KRELIMIT	ANLDAT
1	ABZOCKER, BERLIN	100000,00	27.04.2006

Pseudospalten

Konstante Daten können mit den in der Datenbank gespeicherten Werten kombiniert werden.

Beispiel

```
SELECT ARTIKEL,PREIS1,'EUR' AS WAEHRUNG FROM ART ORDER BY ARTIKEL;
```

ARTIKEL	PREIS1	WAEHRUNG
10.1001	39,99	EUR
10.1016	99,99	EUR
25.3282	4,98	EUR
56.7954	0,40	EUR
80.0001	49,95	EUR

Verknüpfen von Zeichenketten

Ausdrücke des Typs VARCHAR bzw. VARCHAR2 können mittels des Operators || zu einem neuen Ausdruck verknüpft werden.

```
SELECT ARTIKEL,BEZEICHN||' je '||MEH AS BEZEICHN,PREIS1 FROM ART ORDER BY ARTIKEL;
```

ARTIKEL	BEZEICHN	PREIS1
10.1001	Gartenpumpe je Stk	39,99
10.1016	Hochdruckreiniger je Stk	99,99
25.3282	Holzlasur je Lit	4,98
56.7954	Fliesenkleber je Kg	0,40
80.0001	Montageleistungen je Std	49,95

Übungsaufgaben

1. Selektieren Sie alle ausländischen Lieferanten mit den Spalten LIEFNr, KURZNAME und LAND. Die Spalten KURZNAME und LAND sollen zu einer neuen Spalte NAME verknüpft und gemäß nachfolgendem Beispiel ausgegeben werden. Die Sortierung soll aufsteigend nach Lieferantenummer erfolgen.

Ausgabebeispiel:

LIEFNr	NAME
2	MILLER, HAMILTON (BM)
3	LAHO, BAGNEUX (F)
4	FOSTERS, BIRMINGHAM (GB)

2. Selektieren Sie alle Kunden mit den Spalten KUNDNR und KURZNAME, die in den USA ansässig sind und denen ein Kreditlimit zugeordnet ist. Rechnen Sie das Kreditlimit von € in \$ um (1 € = 0,917 \$) und geben Sie den berechneten Wert mit der Spaltenbezeichnung LIMIT im Anschluss an den Kurznamen aus. Außerdem soll die Währungseinheit \$ in einer zusätzlichen Spalte mit dem Namen UNIT ausgegeben werden. Sortieren Sie das Ergebnis absteigend nach der Höhe des eingeräumten Kreditlimits.

Ausgabebeispiel:

KUNDNR	KURZNAME	LIMIT	UNIT
6	BRAD PITT, NY	1,3755	\$
5	BRITNEY SPEARS, LA	0,9170	\$

Lösungen

1. SELECT LIEFNR, KURZNAME || ' (' || LAND || ') ' AS NAME FROM LIE
WHERE LAND IS NOT NULL ORDER BY LIEFNR;
2. SELECT KUNDNR, KURZNAME, KRELIMIT*0.917 AS LIMIT, '\$' AS UNIT FROM KND
WHERE LAND='USA' AND KRELIMIT<>0 ORDER BY KRELIMIT DESC;

Übungsaufgaben

3. Wie Aufgabe 2, allerdings soll das berechnete Kreditlimit incl. Währungssymbol \$ in einer einzigen Spalte mit dem Namen LIMIT ausgegeben werden.

Ausgabebeispiel:

KUNDNR	KURZNAME	LIMIT
6	BRAD PITT, NY	1,3755 \$
5	BRITNEY SPEARS, LA	0,9170 \$

4. Wie Aufgabe 3, allerdings soll das berechnete Kreditlimit auf zwei Nachkommastellen kaufmännisch gerundet werden.

Ausgabebeispiel:

KUNDNR	KURZNAME	LIMIT
6	BRAD PITT, NY	1,38 \$
5	BRITNEY SPEARS, LA	0,92 \$

5. Selektieren Sie sämtliche Kunden mit den Spalten KUNDNR und KURZNAME sowie die Anzahl der Tage, die seit dem Hinzufügen der Kundendaten vergangen sind. Benutzen Sie hierzu die Systemvariablen SYSDATE in ORACLE bzw. 'NOW' in Interbase. Benennen Sie diese Spalte mit dem Namen ANZAHL. Sortieren Sie das Ergebnis absteigend nach der Anzahl der Tage und zusätzlich aufsteigend nach Kundennummer.

Tip: Die Sortierung kann bekanntlich auch nach Spaltennummer, anstatt Spaltennamen, erfolgen!

Ausgabebeispiel:

KUNDNR	KURZNAME	ANZAHL
1	ABZOCKER, BERLIN	16
2	MADGIRLS, LONDON	16
3	STUPIDO, ROMA	14
4	FRAUDEUR, PARIS	14
5	BRITNEY SPEARS, LA	13
6	BRAD PITT, NY	13

6. Ermitteln Sie die tatsächlich verwendeten Länderkennzeichen für Kunden. Die Ausgabe doppelter bzw. leerer Länderkennzeichen soll dabei unterdrückt werden. Sortieren Sie das Ergebnis aufsteigend.

Ausgabebeispiel:

LAND
F
GB
I
USA

Lösungen

3. Oracle:

```
SELECT KUNDNR, KURZNAME, TO_CHAR(KRELIMIT*0.917) || ' $' AS LIMIT  
FROM KND WHERE LAND='USA' AND KRELIMIT <> 0 ORDER BY KRELIMIT DESC;
```

Interbase:

```
SELECT KUNDNR, KURZNAME, CAST(KRELIMIT*0.917 AS VARCHAR(20)) || ' $' AS LIMIT  
FROM KND WHERE LAND='USA' AND KRELIMIT <> 0 ORDER BY KRELIMIT DESC;
```

4. Oracle:

```
SELECT KUNDNR, KURZNAME, TO_CHAR(ROUND(KRELIMIT*0.917, 2)) || ' $' AS LIMIT  
FROM KND WHERE LAND='USA' AND KRELIMIT <> 0 ORDER BY KRELIMIT DESC;
```

Interbase:

```
SELECT KUNDNR, KURZNAME,  
CAST(CAST(KRELIMIT*0.917 AS DECIMAL(10, 2)) AS VARCHAR(20)) || ' $' AS LIMIT  
FROM KND WHERE LAND='USA' AND KRELIMIT <> 0 ORDER BY KRELIMIT DESC;
```

5. Oracle:

```
SELECT KUNDNR, KURZNAME, TRUNC(SYSDATE) - TRUNC(ANLDAT) AS ANZAHL  
FROM KND ORDER BY 3 DESC, KUNDNR;
```

Interbase:

```
SELECT KUNDNR, KURZNAME, CAST('NOW' AS DATE) - CAST(ANLDAT AS DATE) AS ANZAHL  
FROM KND ORDER BY 3 DESC, KUNDNR;
```

6. SELECT DISTINCT LAND FROM KND WHERE LAND IS NOT NULL ORDER BY LAND;