

Studienarbeit

von  
Joachim Müller und Dirk Waldik  
im Fach

Datenbanken und Informationssysteme I  
K532

an der  
Fachhochschule Karlsruhe  
Fachbereich Geoinformationswesen  
Studiengang Kartographie

unter Leitung von  
Prof. Hans Kern

## **Inhaltsverzeichnis:**

Teil 1: Aufgabenblatt

Teil 2: Kurzeinführung in SQL

Teil 3: Handschriftliche Tabellen

Teil 4: Aufgabenstellung und Lösung

Teil 5: Listing mit Dokumentation

Teil 6: Indexverzeichnis

# **Teil 1**

Aufgabenblatt

## **Teil 2**

Kurzeinführung in SQL

## **Teil 3**

Handschriftliche Tabellen

## **Teil 4**

Aufgabenstellung und Lösung



1 row created.  
 1 row created.  
 1 row created.  
 1 row created.  
 1 row created.

**c) Ausgabe der erstellten Tabellen**

**Tabelle 1: karten**

LFD	TITEL	MODUL	BREITE	HÖHE	ZUSTAND	AUSGABE	INHALT	PREIS	LAND
87052	Pforzheim	10000	70	50	KRE	25-AUG-87	Topographie	25.89	BW
89005	Euro Straßen	5000	70	70	Druck	30-SEP-89	Straße	10.99	Hessen
89009	Stuttgart	200000	50	50	Verkauf	01-MAY-89	Touristik	35.2	S-Deutschland
91020	München	100000	60	50	KOH	12-MAR-91	Panorama	5.99	Bayern
92106	Deutschland	1000000	60	60	KRE	18-SEP-92	Topographie	12.59	Deutschland
92308	Saarbrücken	25000	80	70	Planung	19-OCT-92	Panorama	17.5	Saarland
93020	Geo. Berlin	50000	80	50	Druck	20-FEB-93	Spezial	6.99	Berlin
94048	Stadtkarte	16000	20	20	KOH	30-APR-94	Straßen	20.99	Ostfriesland
95001	Hamburg	10000	30	10	Verkauf	10-JAN-95	Topographie	8.9	Hamburg
97042	Mallorca	500000	50	40	Repro	07-JUL-97	Touristik	45.98	Balearen

10 rows selected. ← Zehn Zeilen ausgewählt

**Tabelle 2: abteilungen**

NAME	BUDGET	DECKUNGSBEITRAG	LEITER
KRE-Touristik	200000	180000	Schmidt
KRE-Straßen	500000	650000	Asphalt
KRE-Panorama	380000	300000	Ansicht
KRE-Special	250000	220000	Onga
KOH	750000	600000	Schaber
Repro	190000	250000	Toner
Leitung	80000	80000	Fidel

7 rows selected. ← Sieben Zeilen ausgewählt

**Tabelle 3: mitarbeiter**

NAME	ABTEILUNG	FUNKTION	GEHALTSSTUFE
Humboldt	KRE-Straßen	Geograph	7
Eckert	KOH	Kartograph	5
Lümmeler	KRE-Panorama	Kartograph	6
Lurch	Repro	Drucker	6
Zwecker	KRE-Straßen	Kartograph	7
Kirchenaus	KRE-Special	Graphiker	9
Made	KRE-Touristik	Kartograph	8
Hibiscus	Leitung	Chef	10
Karlsen	Repro	Drucker	4
Integral	KOH	Dipl.-Kartograph	6
Gaius	KRE-Touristik	Künstler	5

11 rows selected. ← Elf Zeilen ausgewählt



**Tabelle 4: erloese**

REGION	KARTEN	ATLANTEN	SONSTIGE	NUTZUNGSRECHTE
Nord-Deutschland	1000	2500	5000	3000
Ost-Deutschland	200	700	2000	50
West-Deutschland	1300	700	1800	375
Süd-Deutschland	5000	8000	40000	4300
Übriges Europa	10000	1586	18275	742
Übrige Welt	500	1250	4723	582

6 rows selected.

← Sechs Zeilen ausgewählt

### Aufgabe 2:

- a) Selektion aller Touristik-Karten, deren Ausgabe nach dem 7. Juli 1997 liegen und einen Maßstab kleiner als 1 : 200 000 sind.

View dropped. [← Falls vorhanden wird die Sicht gelöscht](#)

View created. [← Spalten der Sicht werden erstellt](#)

#### Ausgabe von 'titel', 'modul' und 'preis'

TITEL	MODUL	PREIS
Deutschland	1000000	12.59
Mallorca	500000	45.98

- b) Selektion innerhalb der Tabelle 'mitarbeiter' mit der Bedingung: 'funktion' als Kartograph und 'gehaltsstufe' über sechs.

View dropped. [← Falls vorhanden wird die Sicht gelöscht](#)

View created. [← Spalten der Sicht werden erstellt](#)

#### Ausgabe von 'name', 'abteilung', 'funktion' und 'gehaltsstufe'

NAME	ABTEILUNG	FUNKTION	GEHALTSSTUFE
Zwecker	KRE-Straßen	Kartograph	7
Made	KRE-Touristik	Kartograph	8

### Aufgabe 3:

- a) **Selektion von Karten, die in einem Zeitraum von 27. Dezember 1997 bis 400 Tage davor erschienen sind. Umbenennen der Spalte 'titel' für die Ausgabe in den 'titel der karte'.**

View dropped. [← Falls vorhanden wird die Sicht gelöscht](#)  
View created. [← Spalten der Sicht werden erstellt](#)

#### Ausgabe von 'ausgabe' und 'titel der ausgabe'

```
AUSGABE  TITEL DER AUSGABE  
-----  
07-JUL-97 Mallorca
```

- b) **Berechnung des Kartenpreises in Dollar (Kurspreis: 1 DM = 1,41 \$) und Versandanteil von 0,4 DM pro Karte.**

View dropped. [← Falls vorhanden wird die Sicht gelöscht](#)  
View created. [← Spalten der Sicht werden erstellt](#)

#### Ausgabe des Dollarpreises mit zwei Nachkommastellen.

```
TITEL          PREIS IN DM  PREIS IN DOLLAR  PREIS IN DOLLAR INKL. VERSAND  
-----  
Pforzheim      25.89         36.5             37.07  
Euro Straßen   10.99         15.5             16.06  
Stuttgart      35.2          49.63            50.2  
München        5.99          8.45             9.01  
Deutschland    12.59         17.75            18.32  
Saarbrücken   17.5          24.68            25.24  
Geo. Berlin    6.99          9.86             10.42  
Stadtkarte     20.99         29.6             30.16  
Hamburg        8.9           12.55            13.11  
Mallorca       45.98         64.83            65.4
```

10 rows selected. [← Zehn Zeilen ausgewählt](#)

- c) **Berechnung der Summe der regionalisierten Erlöse und der prozentualen Anteile der 'karten', 'atlanten', 'sonstige' und 'nutzungsrechte'. Ermittlung der Anfangswinkel von 'karten', 'atlanten', 'sonstige' und 'nutzungsrechte' für die Kreisdiagramme der regionalisierten Erlöse.**

Table altered. [← Erweitern der Tabelle 'erloese' um die Spalte 'reg\\_sum'](#)  
1 row updated. [← Zeilenweises hinzufügen der neuen Daten in Spalte 'reg\\_sum'](#)  
1 row updated.  
1 row updated.  
1 row updated.  
1 row updated.  
1 row updated.  
1 row updated.  
Table altered. [← Erweitern der Tabelle 'erloese' um eine neue Spalte 'reg\\_nr'](#)

1 row updated. ← Zeilenweises hinzufügen der neuen Daten in Spalte 'reg\_nr'  
 1 row updated.  
 1 row updated.  
 1 row updated.  
 1 row updated.  
 1 row updated.

**Ausgabe der Summe der regionalisierten Erlöse**

GRUPPEN	GESAMTER ERLOES	JE GRUPPE
1	75925	
2	30603	
3	7055	

**Ausgabe des Gesamterloeses**

GESAMTERLOES
113583

**Ausgabe des prozentualen Anteils je Region**

GESAMTERLOES JE REGION	KARTEN	ATLANTEN	SONSTIGE	NUTZUNGSRECHTE
11500	8.7	21.74	43.48	26.09
2950	6.78	23.73	67.8	1.69
4175	31.14	16.77	43.11	8.98
57300	8.73	13.96	69.81	7.5
30603	32.68	5.18	59.72	2.42
7055	7.09	17.72	66.95	8.25

6 rows selected. ← Sechs Zeilen ausgewählt

**Ausgabe der Gruppensumme der regionalisierten Erlöse in absoluten Werten**

GESAMTERLOES JE GRUPPE	KARTEN	ATLANTEN	SONSTIGE	NUTZUNGSRECHTE
75925	7500	11900	48800	7725
30603	10000	1586	18275	742
7055	500	1250	4723	582

**Ausgabe der Gruppensumme der regionalisierten Erlöse in Prozent**

GESAMTERLOES JE GRUPPE	KARTEN	ATLANTEN	SONSTIGE	NUTZUNGSRECHTE
75925	9.88	15.67	64.27	10.17
30603	32.68	5.18	59.72	2.42
7055	7.09	17.72	66.95	8.25

### Ausgabe des prozentualen Anteils an dem Gesamterloes

GESAMTERLOES	KARTEN	ATLANTEN	SONSTIGE	NUTZUNGSRECHTE
113583	15.85	12.97	63.21	7.97

### Ausgabe der Winkel-Wert der Produkte je Gruppe

GESAMTERLOES JE GRUPPE	KARTEN	ATLANTEN	SONSTIGE	NUTZUNGSRECHTE
75925	35.56	56.42	231.39	36.63
30603	117.64	18.66	214.98	8.73
7055	25.51	63.78	241	29.7

**Aufgabe 4:**

a) Erstellen einer Liste, sortiert nach 'abteilung', aus den Tabellen 'mitarbeiter' und 'abteilungen'.

View dropped. ← Falls vorhanden wird die Sicht gelöscht

View created. ← Spalten der Sicht werden erstellt

**Ausgabe der Spalten 'mitarbeitername', 'funktion', 'budget', 'leiter' und 'gehaltsstufe', sortiert nach 'abteilung'.**

ABTEILUNG GEHALTSSTUFE	MITARBEITERNAME	FUNKTION	BUDGET	LEITER
-				
KOH 5	Eckert	Kartograph	750000	Schaber
KOH 6	Integral	Dipl.-Kartograph	750000	Schaber
KRE-Panorama 6	Lümmler	Kartograph	380000	Ansicht
KRE-Special 9	Kirchenaus	Graphiker	250000	Onga
KRE-Straßen 7	Humbold	Geograph	500000	Asphalt
KRE-Straßen 7	Zwecker	Kartograph	500000	Asphalt
KRE-Touristik 8	Made	Kartograph	200000	Schmidt
KRE-Touristik 5	Gaius	Künstler	200000	Schmidt
Leitung 10	Hibiscus	Chef	80000	Fidel
Repro 6	Lurch	Drucker	190000	Toner
Repro 4	Karlsen	Drucker	190000	Toner

11 rows selected. ← Elf Zeilen ausgewählt

b) Erstellen einer Liste aller Karten, die eine Kartenfläche haben die kleiner ist als die Karte von Saarbrücken (ohne die Fläche der Vergleichskarte zu ermitteln).

View dropped. ← Falls vorhanden wird die Sicht gelöscht

View created. ← Spalten der Sicht werden erstellt

**Ausgabe der Karten aus der Tabelle 'karten' (Spalte 'titel')**

```
TITEL
-----
Pforzheim
Euro Straßen
Stuttgart
München
Deutschland
Geo. Berlin
Stadtkarte
Hamburg
Mallorca
```

9 rows selected. ← Elf Zeilen ausgewählt

**Aufgabe 5:**

- a) Erstellen einer Liste mit der Anzahl der Touristik- und Panoramakarten, sowie deren durchschnittlichen Preis.

View dropped. [← Falls vorhanden wird die Sicht gelöscht](#)  
View created. [← Spalten der Sicht werden erstellt](#)

**Ausgabe der Anzahl an Karten aus der Tabelle 'karten'**

ANZAHL	DURCHSCHNITTLICHER PREIS
2	11,75
2	40,59

- b) Gruppierung der Karten nach Preisklassen durch Einfügen der Spalte 'preisklasse' in die Tabelle 'karten' mit einem Befehl (die Preisklassen sind dabei in 5.- DM - Schritten eingeteilt).

Table altered. [← Erweitern der Tabelle 'karten' um die Spalten 'von', 'bis unter DM' und 'anzahl Karten pro preisklasse'](#)  
10 rows selected. [← Zehn Zeilen ausgewählt](#)

**Ausgabe der Preisklassen, deren Bereiche, sowie die Anzahl der Karten pro Preisklasse**

PREISKLASSE	VON	BIS	UNTER	DM	ANZAHL KARTEN PRO PK
2		5		10	3
3		10		15	2
4		15		20	1
5		20		25	1
6		25		30	1
8		35		40	1
10		45		50	1

7 rows selected. [← Sieben Zeilen ausgewählt](#)

- c) Siehe Aufgabe 3c)

**Aufgabe 6:**

**Generierung von Views der Tabellen ohne den Spalten nach eigener Wahl, mit entsprechender Abfrage nach jedem View.**

View dropped. ← Falls vorhanden wird die Sicht gelöscht

View created. ← Spalten der Sicht werden erstellt

**View der Tabelle 'karten'**

LFD	TITEL	MODUL	BREITE	HÖHE	ZUSTAND	AUSGABE	INHALT	PREIS
87052	Pforzheim	10000	70	50	KRE	25-AUG-87	Topographie	25.89
89005	Euro Straßen	5000	70	70	Druck	30-SEP-89	Straße	10.99
89009	Stuttgart	200000	50	50	Verkauf	01-MAY-89	Touristik	35.2
91020	München	100000	60	50	KOH	12-MAR-91	Panorama	5.99
92106	Deutschland	1000000	60	60	KRE	18-SEP-92	Topographie	12.59
92308	Saarbrücken	25000	80	70	Planung	19-OCT-92	Panorama	17.5
93020	Geo. Berlin	50000	80	50	Druck	20-FEB-93	Spezial	6.99
94048	Stadtkarte	16000	20	20	KOH	30-APR-94	Straßen	20.99
95001	Hamburg	10000	30	10	Verkauf	10-JAN-95	Topographie	8.9
97042	Mallorca	500000	50	40	Repro	07-JUL-97	Touristik	45.98

10 rows selected. ← Zehn Zeilen ausgewählt

View dropped. ← Falls vorhanden wird die Sicht gelöscht

View created. ← Spalten der Sicht werden erstellt

**View der Tabelle 'abteilungen'**

NAME	BUDGET	DECKUNGSBEITRAG
KRE-Touristik	200000	180000
KRE-Straßen	500000	650000
KRE-Panorama	380000	300000
KRE-Special	250000	220000
KOH	750000	600000
Repro	190000	250000
Leitung	80000	80000

7 rows selected. ← Sieben Zeilen ausgewählt

View dropped. ← Falls vorhanden wird die Sicht gelöscht

View created. ← Spalten der Sicht werden erstellt



### View der Tabelle 'mitarbeiter'

NAME	ABTEILUNG	FUNKTION
Humbold	KRE-Straßen	Geograph
Eckert	KOH	Kartograph
Lümmler	KRE-Panorama	Kartograph
Lurch	Repro	Drucker
Zwecker	KRE-Straßen	Kartograph
Kirchemaus	KRE-Special	Graphiker
Made	KRE-Touristik	Kartograph
Hibiscus	Leitung	Chef
Karlsen	Repro	Drucker
Integral	KOH	Dipl.-Kartograph
Gaius	KRE-Touristik	Künstler

11 rows selected.

← Elf Zeilen ausgewählt

View dropped.

← Falls vorhanden wird die Sicht gelöscht

View created.

← Spalten der Sicht werden erstellt

### View der Tabelle 'erloese'

REGION	KARTEN	ATLANTEN	SONSTIGE
Nord-Deutschland	1000	2500	5000
Ost-Deutschland	200	700	2000
West-Deutschland	1300	700	1800
Süd-Deutschland	5000	8000	40000
Übriges Europa	10000	1586	18275
Übrige Welt	500	1250	4723

6 rows selected.

← Sechs Zeilen ausgewählt

## **Teil 5**

Listing mit Dokumentation

### Anmerkungen:

Die in COURIER geschriebenen Worte sind identisch zu den Befehlen oder Bezeichnungen aus den Listings. Vorgegebene feststehende Befehle unter SQL sind in GROSSBUCHSTABEN geschrieben und müssen mit einem Semikolon abgeschlossen werden.

Alle in blau geschriebenen Worte dienen zur Beschreibung der Funktionalität der einzelnen Befehle und beschreibt deren Wirkung. Zusätzlich werden in blau Anmerkungen, Hinweise und Begründungen an die einzelnen Aufgaben angehängt.

Die in *kursiv* geschriebenen Worte entsprechen den Namen der Tabellen und der Spalten.

Das Ergebnis der Eingaben, Befehle und der Resultate ist im Anschluß der Aufgabenstellung als Listing (erstellt durch eine SPOOL-Datei, s. unten) mit zusätzlicher Dokumentation beigefügt.

### Vorbereitungen:

#### a) Erstellen einer START- und SPOOL-Datei:

Für die einfach Handhabung von SQL-Datenbanken wird eine sogenannte START-Datei erstellt. Diese Datei enthält den Befehl ein Listing zu erstellen, das mit dem Befehl < SPOOL 'Laufwerk:Dateiname.lst'; > realisiert. Die entstandene SPOOL -Datei (siehe Teil 2 der Ausarbeitung) enthält alle Ausführungen, die durch die einzelnen Befehle aufgerufen werden. Sie dient somit zur Ergebnisausgabe (z. B. auf Bildschirm oder Papier) und gerade bei größeren Datenbanken zu einem überschaulichen Hilfsmittel.

Formatierungen für die Typographie, die für die Ausgabe der ganzen Datenbank gelten, können hier global definiert werden.

Das besondere Merkmal einer START-Datei ist aber das einzelne Aufrufen von SQL-Dateien., d. h. um die Übersicht einer großen Datenbank zu bewahren können hier einzelne Teilbereiche (erstellen von Tabellen, Abfragen über einzelne/mehrere Tabellen, Sichte, ...) aufgerufen werden. Mit der als Zentrale wirkende START-Datei kann außerdem die komplette Datenbank mit einem Aufruf erstellt werden.

```
SPOOL a:\my_sql.lst;           ←SPOOL-Datei 'my_sql.lst' wird auf Laufwerk a: erstellt.

SET LINESIZE 300              ←Linienlänge auf 300 Zeichen
SET PAGESIZE 66              ←Linien pro Seite auf 66

@a:\karten;                  ←Aufruf der Dateien zum Erstellen der Tabellen
@a:\abteilu;
@a:\mitarb;
@a:\erloese;

SELECT * FROM karten;        ←Auswahl von allen (*) Spalten der jeweiligen Tabelle,
SELECT * FROM abteilungen;   d.h. die oben erstellten Tabellen werden komplett am
SELECT * FROM mitarbeiter;    Bildschirm dargestellt.
SELECT * FROM erloese;

/* Aufgabe 2 */              ←/* ... dies ist ein Kommentar ... */

@a:\aufg2;                  ←Aufruf der einzelnen Dateien nach Aufgaben unterteilt.

/* Aufgabe 3 */

@a:\aufg3;

/* Aufgabe 4 */

@a:\aufg4;

/* Aufgabe 5 */

@a:\aufg5;

/* Aufgabe 6 */

@a:\aufg6;

SPOOL OFF;                  ←Ende der SPOOL-Datei
```

## Aufgabe 1:

### a) b) Erstellen der Tabellen durch definieren der Spalten und zeilenweises eintragen der Daten.

**Hinweis: Die folgenden vier Tabellen der Aufgabe 1 sind Grundlage der Aufgaben 2-6 !**

```
DROP TABLE karten;
```

←DROP TABLE löscht, falls vorhanden, die Tabelle 'karten'.

```
CREATE TABLE karten (  
    lfd NUMBER(5),  
    titel CHAR(20),  
    modul NUMBER(7),  
    breite NUMBER(2),  
    höhe NUMBER(2),  
    zustand CHAR(10),  
    ausgabe DATE,  
    inhalt CHAR(15),  
    preis NUMBER(6,2),  
    land CHAR(20));
```

←Die Tabelle 'karten' wird erstellt, indem die einzelnen Spalten durch Name und Datentyp (mit Parametern) definiert werden.

DATENTYPEN:

- NUMBER(a, b)

entspricht einer Ziffer mit a Stellen und b Stellen hinter dem Komma

- CHAR(a)

entspricht einer Zeichenkette mit a Zeichen

- DATE

entspricht einem Datentyp zur Eingabe des Datums

• INSERT INTO *tabelle* VALUES(Daten)

Zeilenweise Eingabe der Daten in die *tabelle* nach der Reihenfolge entsprechend der vorher definierten Spalten (CREATE TABLE).

Hinweise:- Textstrings müssen in Hochkommas stehen

- Das Datum muß mit englischem Monatsname und Hochkommas angegeben werden ('01-MAY-99')

```
INSERT INTO karten VALUES  
    (87052, 'Pforzheim', 10000, 70, 50, 'KRE', '25-AUG-87', 'Topographie', 25.89, 'BW');  
  
INSERT INTO karten VALUES  
    (89005, 'Euro Straßen', 5000, 70, 70, 'Druck', '30-SEP-89', 'Straße', 10.99, 'Hessen');  
  
INSERT INTO karten VALUES  
    (89009, 'Stuttgart', 200000, 50, 50, 'Verkauf', '01-MAY-89', 'Touristik', 35.20, 'S-Deutschland');  
  
INSERT INTO karten VALUES  
    (91020, 'München', 100000, 60, 50, 'KOH', '12-MAR-91', 'Panorama', 5.99, 'Bayern');  
  
INSERT INTO karten VALUES  
    (92106, 'Deutschland', 1000000, 60, 60, 'KRE', '18-SEP-92', 'Topographie', 12.59, 'Deutschland');  
  
INSERT INTO karten VALUES  
    (92308, 'Saarbrücken', 25000, 80, 70, 'Planung', '19-OCT-92', 'Panorama', 17.50, 'Saarland');  
  
INSERT INTO karten VALUES  
    (93020, 'Geo. Berlin', 50000, 80, 50, 'Druck', '20-FEB-93', 'Spezial', 6.99, 'Berlin');  
  
INSERT INTO karten VALUES  
    (94048, 'Stadtkarte', 16000, 20, 20, 'KOH', '30-APR-94', 'Straßen', 20.99, 'Ostfriesland');  
  
INSERT INTO karten VALUES  
    (95001, 'Hamburg', 10000, 30, 10, 'Verkauf', '10-JAN-95', 'Topographie', 8.90, 'Hamburg');  
  
INSERT INTO karten VALUES  
    (97042, 'Mallorca', 500000, 50, 40, 'Repro', '07-JUL-97', 'Touristik', 45.98, 'Balearen');
```

↑

• Es werden 10 Datensätze/Zeilen eingefügt

Das Erstellen der folgenden Tabellen *'abteilungen'*, *'mitarbeiter'* und *'erloese'* und das Einfügen der Daten erfolgt nun äquivalent zu dem Verfahren der obigen Tabelle *'karten'*.

#### Tabelle *'abteilungen'*

```
DROP TABLE abteilungen;  
  
CREATE TABLE abteilungen (  
    name CHAR(15),  
    budget NUMBER(10),  
    deckungsbeitrag NUMBER(10),  
    leiter CHAR(10));  
  
INSERT INTO abteilungen VALUES('KRE-Touristik',200000,180000,'Schmidt');  
INSERT INTO abteilungen VALUES('KRE-Straßen',500000,650000,'Asphalt');  
INSERT INTO abteilungen VALUES('KRE-Panorama',380000,300000,'Ansicht');  
INSERT INTO abteilungen VALUES('KRE-Special',250000,220000,'Onga');  
INSERT INTO abteilungen VALUES('KOH',750000,600000,'Schaber');  
INSERT INTO abteilungen VALUES('Repro',190000,250000,'Toner');  
INSERT INTO abteilungen VALUES('Leitung',80000,80000,'Fidel');
```



- Es werden 7 Datensätze/Zeilen eingefügt

#### Tabelle *'mitarbeiter'*

```
DROP TABLE mitarbeiter;  
  
CREATE TABLE mitarbeiter (  
    name CHAR(20),  
    abteilung CHAR(20),  
    funktion CHAR(20),  
    gehaltsstufe NUMBER(5));  
  
INSERT INTO mitarbeiter VALUES ('Humbold','KRE-Straßen','Geograph',7);  
INSERT INTO mitarbeiter VALUES ('Eckert','KOH','Kartograph',5);  
INSERT INTO mitarbeiter VALUES ('Lümmler','KRE-Panorama','Kartograph',6);  
INSERT INTO mitarbeiter VALUES ('Lurch','Repro','Drucker',6);  
INSERT INTO mitarbeiter VALUES ('Zwecker','KRE-Straßen','Kartograph',7);  
INSERT INTO mitarbeiter VALUES ('Kirchenaus','KRE-Special','Graphiker',9);  
INSERT INTO mitarbeiter VALUES ('Made','KRE-Touristik','Kartograph',8);  
INSERT INTO mitarbeiter VALUES ('Hibiscus','Leitung','Chef',10);  
INSERT INTO mitarbeiter VALUES ('Karlsen','Repro','Drucker',4);  
INSERT INTO mitarbeiter VALUES ('Integral','KOH','Dipl.-Kartograph',6);  
INSERT INTO mitarbeiter VALUES ('Gaius','KRE-Touristik','Künstler',5);
```



- Es werden 11 Datensätze/Zeilen eingefügt

### Tabelle 'erloese'

```
DROP TABLE erloese;
```

```
CREATE TABLE erloese (  
    region CHAR(20),  
    karten NUMBER(6),  
    atlanten NUMBER(6),  
    sonstige NUMBER(6),  
    nutzungsrechte NUMBER(6));
```

```
INSERT INTO erloese VALUES('Nord-Deutschland',1000,2500,5000,3000);
```

```
INSERT INTO erloese VALUES('Ost-Deutschland',200,700,2000,50);
```

```
INSERT INTO erloese VALUES('West-Deutschland',1300,700,1800,375);
```

```
INSERT INTO erloese VALUES('Süd-Deutschland',5000,8000,40000,4300);
```

```
INSERT INTO erloese VALUES('Übriges Europa',10000,1586,18275,742);
```

```
INSERT INTO erloese VALUES('Übrige Welt',500,1250,4723,582);
```



- Es werden 6 Datensätze/Zeilen eingefügt

## Aufgabe 2:

- a) **Touristik-Karten, deren Ausgabe nach einem beliebigen Datum ('1-MAY-89)liegen und die einen Maßstab kleiner als 1 : 200 000 haben, sollen selektiert und ausgegeben werden. Ausgegeben werden dabei die Spalten 'titel', 'maßstab' und 'preis'**

```
/*  
Aufgabe 2a */
```

←Kommentar

```
DROP VIEW aufgabe2a;
```

←DROP VIEW löscht, falls vorhanden, den View 'aufgabe2a'.

- CREATE VIEW 'viewname' AS kreiert eine Sicht auf bereits bestehende Tabellen durch einen SELECT-Befehl. Diese Sicht ist eine „virtuelle“ Tabelle, die wie eine normale Tabelle benutzt werden kann. Diese Sicht ist zum Unterschied zu einer Tabelle von der Tabelle abhängig, von der sie die Daten bekommt, d. h. wenn die Tabelle verändert wird, verändert sich die Sicht bei ihrem Aufruf ebenfalls.  
Hier wird eine Sicht 'aufgabe2a' erstellt, die die Spalten 'titel', 'modul' und 'preis' von der Tabelle 'karten' enthält und die den Bedingungen (WHERE...AND...) des SELECT-Befehls genügen.

```
CREATE VIEW aufgabe2a AS  
SELECT titel, modul, preis FROM karten WHERE inhalt='Touristik' AND modul>200000 AND ausgabe  
BETWEEN '01-MAY-89' AND '07-JUL-97';
```

↑

- BETWEEN...AND... ist eine Abfrage nach dem Bereich zwischen zwei Parametern gleichen Typs.

```
SELECT * FROM aufgabe2a;
```

←Es werden alle (\*) Datensätze/Zeilen von der Sicht 'aufgabe2a' ausgewählt bzw. auf dem Bildschirm ausgegeben.

- b) **Auswahl aller in der Tabelle 'mitarbeiter' enthaltenen Mitarbeiter, die eine Gehaltsstufe über sechs und die Funktion als Kartograph im kartographischen Verlag haben**

```
/*  
Aufgabe 2b */
```

←Kommentar

```
DROP VIEW aufgabe2b;
```

←DROP VIEW löscht, falls vorhanden, den View 'aufgabe2b'.

```
CREATE VIEW aufgabe2b AS
```

←CREATE VIEW 'aufgabe2b' AS kreiert die Sicht 'aufgabe2b'.

- Auswahl aller (\*) Datensätze aus der Tabelle 'mitarbeiter' mit der Bedingung WHERE...AND... (s. Aufgabenstellung).

```
SELECT * FROM mitarbeiter WHERE gehaltsstufe>6 AND funktion='Kartograph';
```

```
SELECT * FROM aufgabe2b;
```

←Es werden alle (\*) Datensätze/Zeilen von der Sicht 'aufgabe2a' ausgewählt bzw. auf dem Bildschirm ausgegeben.  
Ausgabe von 'name', 'abteilung', 'funktion' und 'gehaltsstufe'.

### Aufgabe 3:

- a) **Selektion von Karten, die in einem Zeitraum von 27. Dezember 1997 bis 400 Tage davor erschienen sind. Bei der Ausgabe soll die Spalte 'titel' in den Spaltennamen 'titel der karte' umbenannt werden**

```
/*                                     ←Kommentar
Aufgabe 3a */

DROP VIEW aufgabe3a;                 ←DROP VIEW löscht, falls vorhanden, den View 'aufgabe3a'.

CREATE VIEW aufgabe3a AS             ←CREATE VIEW 'aufgabe3a' AS kreiert die Sicht 'aufgabe3a'.

                                     • BETWEEN ADD MONTHS('datum',a) AND 'datum' stellt ein
                                     | Datum zwischen dem Zeitraum als Bedingung, dabei ist a die Zahl
                                     | der Monate die auf (ADD_MONTHS('datum',a)) das 'datum'
                                     | addiert bzw. bei negativem Wert subtrahiert werden
                                     | (hier: 400/30=13 Monate).
                                     ↓
SELECT  ausgabe, titel "titel der karte" FROM karten WHERE ausgabe BETWEEN ADD_MONTHS('27-
DEC-97',-400/30) AND '27-DEC-97';    ↑

                                     • Wenn der Name einer Spalte bei der Ausgabe einen anderen
                                     Namen haben soll, wird der neue Name in Anführungszeichen
                                     hinter dem alten Namen geschrieben.

SELECT * FROM aufgabe3a;             ←Es werden alle (*) Datensätze/Zeilen von der Sicht 'aufgabe3a'
                                     ausgewählt bzw. auf dem Bildschirm ausgegeben.
                                     Ausgabe von 'ausgabe' und 'titel der karte'.
```

- b) **Berechnung des Kartenpreises mit zwei Nachkommastellen in Dollar (Kurspreis: 1 DM = 1,41 \$) und eines Versandanteils von 0,4 DM pro Karte. Der Titel soll auf 12 Zeichen gekürzt werden**

```
/*                                     ←Kommentar
Aufgabe 3b */

DROP VIEW aufgabe3b;                 ←DROP VIEW löscht, falls vorhanden, den View 'aufgabe3b'.

CREATE VIEW aufgabe3b AS             ←CREATE VIEW 'aufgabe3a' AS kreiert die Sicht 'aufgabe3b'.

                                     • ROUND('spalte',a) rundet die Werte einer 'spalte' mit a Nach-
                                     | kommastellen. Durch Arithmetische Ausdrücke mit den
                                     | Operatoren +,-,* und / können dabei die Werte der jeweiligen
                                     | Spalte verändert werden.
                                     ↓
SELECT  titel, preis "preis in DM", ROUND(1.41*preis,2) "preis in US$",
ROUND((preis+0.40)*1.41,2) "preis in US$ inkl. versand" FROM karten;    ↑

                                     • Wenn Spalten durch arithmetische Ausdrücke verändert werden,
                                     muß eine neue Spalte mit anderem Spaltennamen angelegt
                                     werden. Der Namen der Spalte 'preis' wird zu 'preis in DM'
                                     umbenannt. Der berechnete Kartenpreis in Dollar (berechnet aus
                                     Spalte 'preis') wird in 'preis in US$' und der Versandteil
                                     (ebenfalls berechnet aus Spalte 'preis') benannt.

COLUMN titel FORMAT A12;             ←Setzt die Spalte 'titel' auf die Länge 12 Zeichen .

SELECT * FROM aufgabe3b;             ←Es werden alle (*) Datensätze/Zeilen von der Sicht 'aufgabe3a' .
                                     ausgewählt bzw. auf dem Bildschirm ausgegeben. Ausgabe von
                                     'preis in US$', 'preis in US$' und 'preis in US$ inkl. versand'.
```



c) **Berechnung der Summe der regionalisierten Erlöse und der prozentualen Anteile der 'karten', 'atlanten', 'sonstige' und 'nutzungsrechte'. Ermittlung der Anfangswinkel von 'karten', 'atlanten', 'sonstige' und 'nutzungsrechte' für die Kreissektorendiagramme der regionalisierten Erlöse.**

/\*  
Aufgabe 3c \*/

←Kommentar

1. Schritt: Bildung der Summe der regionalisierten Erlöse (vier Spalten) für die jeweilige Regionen.

Die Zeilen werden hier der Reihe nach der 'region' aufgezählt und die Summe in eine neue Spalte eingefügt, man erhält die Summe der regionalisierten Erlöse.

- Durch den Befehl ALTER TABLE 'tabelle' ADD ('spalte' Datentyp) wird eine neue Spalte (mit definition des Datentyps) an die Tabelle 'tabelle' angefügt.  
Hier wird der Tabelle 'erloese' die Spalte 'reg\_sum' mit maximal zehnstelligen Ziffern erweitert.

ALTER TABLE erloese ADD (reg\_sum NUMBER(10));

- Die neue Spalte 'reg\_sum' der Tabelle 'erloese' wird nun mit den entsprechenden Daten aufgefüllt.  
Hier wird gleichzeitig die Summe der vier Spalten 'karten', 'atlanten', 'sonstige' und 'nutzungsrechte' gebildet und in die entsprechende Zelle der neuen Spalte gesetzt, wo nach Bedingung (WHERE) erfüllt ist, d. h. in der Spalte 'region' muß die jeweilige Region stehen.

```
UPDATE erloese SET
    reg_sum= karten+atlanten+sonstige+nutzungsrechte WHERE region='Nord-Deutschland';
UPDATE erloese SET
    reg_sum= karten+atlanten+sonstige+nutzungsrechte WHERE region='Ost-Deutschland';
UPDATE erloese SET
    reg_sum= karten+atlanten+sonstige+nutzungsrechte WHERE region='West-Deutschland';
UPDATE erloese SET
    reg_sum= karten+atlanten+sonstige+nutzungsrechte WHERE region='Süd-Deutschland';
UPDATE erloese SET
    reg_sum= karten+atlanten+sonstige+nutzungsrechte WHERE region='Übriges Europa';
UPDATE erloese SET
    reg_sum= karten+atlanten+sonstige+nutzungsrechte WHERE region='Übrige Welt';
```

2. Schritt: Bildung der Gruppen Deutschland, übriges Europa und übrige Welt.

Zur Klassifizierung müssen sogenannte Kennziffer gebildet werden.

- (siehe oben)

ALTER TABLE erloese ADD (reg\_nr NUMBER(10));

- Die neue Spalte 'reg\_nr' der Tabelle 'erloese' wird mit den drei Gruppenkennziffern aufgefüllt.

```
UPDATE erloese SET reg_nr=1 WHERE region='Nord-Deutschland';
UPDATE erloese SET reg_nr=1 WHERE region='Ost-Deutschland';
UPDATE erloese SET reg_nr=1 WHERE region='West-Deutschland';
UPDATE erloese SET reg_nr=1 WHERE region='Süd-Deutschland';
UPDATE erloese SET reg_nr=2 WHERE region='Übriges Europa';
UPDATE erloese SET reg_nr=3 WHERE region='Übrige Welt';
```

```
/*                               ←Kommentar  
Summe der regionalisierten Erlöse */
```

- Durch GROUP BY findet eine Auswahl entsprechend der Gruppenkennziffern statt, d. h. die Zeilen mit der selben Gruppenkennziffer werden zusammen ausgewertet und ausgegeben.  
HAVING entspricht der Funktionalität von WHERE, ist aber bei der Auswahl-Bedingung für Gruppen mit HAVING und für Zeilen mit WHERE.

```
SELECT reg_nr "gruppen", SUM(reg_sum) "gesamter erloes je gruppe" FROM erloese GROUP BY reg_nr  
HAVING reg_nr<4;
```

- Es werden die Gruppenkennziffern als Spaltenname 'gruppen' ausgegeben. Die Summe (SUM(*spalte*)) der regionalisierten Erlöse entsprechend der Gruppenbildung mit den Gruppenkennziffern wird unter 'gesamter erloes je gruppe' ausgegeben.

```
SELECT SUM(reg_sum) "gesamterloes" FROM erloese;
```

- Ausgabe des Gesamterlöses durch die Addition der Spalte 'reg\_sum' von der Tabelle 'erloese'.

```
/*  
%-Anteil je Region */
```

- Zur weiteren Information werden die prozentualen Anteile der Erlöse je Region für die einzelnen Spalten ausgegeben  
Die prozentualen Werte werden hier durch die Spalte und dem jeweiligen regionalisierten Erlös berechnet (Dreisatz) und auf zwei Stellen hinter dem Komma gerundet (ROUND(...,2)).

```
SELECT reg_sum "gesamterloes je region",  
ROUND(karten*100/reg_sum,2) "karten",  
ROUND(atlanten*100/reg_sum,2) "atlanten",  
ROUND(sonstige*100/reg_sum,2) "sonstige",  
ROUND(nutzungsrechte*100/reg_sum,2) "nutzungsrechte"  
FROM erloese;
```

- Ausgabe des Gesamterlöses und der prozentualen Erlöse je Region.

```
/*  
Gruppensumme der regionalisierten Erlöse in absoluten Werten */
```

- Zur weiteren Information werden die absoluten Anteile der regionalisierten Erlöse (durch Gruppenbildung) berechnet.  
Die Gruppenbildung entsteht dabei durch die Spalte 'reg\_nr'  
Die Bedingung für die Gruppenbildung ist hier durch HAVING auf Gruppennummer unter vier gesetzt.

```
SELECT SUM(reg_sum) "gesamterloes je gruppe", SUM(karten) "karten",  
SUM(atlanten) "atlanten",  
SUM(sonstige) "sonstige",  
SUM(nutzungsrechte) "nutzungsrechte"  
FROM erloese GROUP BY reg_nr HAVING reg_nr<4;
```

- Ausgabe des Gesamterlöses und der absoluten Erlöse je Gruppe.

```
/*  
Gruppensumme der regionalisierten Erlöse in Prozent*/
```

- Zur Aufgabenstellung werden hier die prozentualen Anteile der regionalisierten Erlöse (durch Gruppenbildung) berechnet. Die Gruppenbildung entsteht dabei durch die Spalte 'reg\_nr'. Die Bedingung für die Gruppenbildung ist hier durch HAVING auf Gruppennummer unter vier gesetzt.

```
SELECT SUM(reg_sum) "gesamterloes je Gruppe",  
ROUND(SUM(karten)*100/SUM(reg_sum),2) "karten",  
ROUND(SUM(atlanten) *100/SUM(reg_sum),2) "atlanten",  
ROUND(SUM(sonstige) *100/SUM(reg_sum),2) "sonstige",  
ROUND(SUM(nutzungsrechte) *100/SUM(reg_sum),2)  
"nutzungsrechte"  
FROM erloese GROUP BY reg_nr HAVING reg_nr<4;
```

- Ausgabe des Gesamterlöses und der prozentualen Erlöse je Gruppe.

```
/*  
%-Anteil an dem Gesamterloes */
```

- Zur weiteren Information werden die prozentualen Anteile der summierten regionalisierten Erlöse berechnet, d.h. der Anteil der 'karten', 'atlanten', 'sonstige' und 'nutzungsrechte' am Gesamterlös.

```
SELECT SUM(reg_sum) "Gesamterloes",  
ROUND(SUM(karten)*100/SUM(reg_sum),2) "karten",  
ROUND(SUM(atlanten) *100/SUM(reg_sum),2) "atlanten",  
ROUND(SUM(sonstige) *100/SUM(reg_sum),2) "sonstige",  
ROUND(SUM(nutzungsrechte) *100/SUM(reg_sum),2)  
"nutzungsrechte"  
FROM erloese;
```

- Ausgabe des Gesamterlöses und der prozentualen Erlöse der Spalten.

```
/*  
Winkelwert der Produkte je Gruppe */
```

- Die zu berechnenden Winkel werden wie die prozentualen Anteile berechnet, nur mit dem Faktor 360° (Vollkreis).

```
SELECT SUM(reg_sum) "gesamterloes je gruppe",  
ROUND(SUM(karten)*360/SUM(reg_sum),2) "karten",  
ROUND(SUM(atlanten) *360/SUM(reg_sum),2) "atlanten",  
ROUND(SUM(sonstige) *360/SUM(reg_sum),2) "sonstige",  
ROUND(SUM(nutzungsrechte) *360/SUM(reg_sum),2)  
"nutzungsrechte"  
FROM erloese GROUP BY reg_nr HAVING reg_nr<4;
```

- Ausgabe des Gesamterlöses und der entsprechenden Winkel für einen Vollkreis je regionalisierter Erlöse.

#### Aufgabe 4:

a) Erstellen einer Liste, sortiert nach 'abteilung', aus den Tabellen 'mitarbeiter' und 'abteilungen'.

```
/*  
Aufgabe 4a */
```

```
DROP VIEW aufgabe4a;
```

←DROP VIEW löscht, falls vorhanden, die Sicht 'aufgabe4a'.

```
CREATE VIEW aufgabe4a AS
```

←CREATE VIEW 'aufgabe4a' AS kreiert die Sicht 'aufgabe4a'.

- Für die Abfrage nach Daten in verschiedenen Tabellen wird eine verbundene Auswahl (join) aufgerufen. Die verbundene Auswahl wird durch eine WHERE-Bedingung realisiert und am Ende des SELECT-Befehls angehängt. Sie verknüpft zwei Tabellen die ein und dieselbe Spalte haben. Die Tabellen mit der gemeinsamen Spalte (mit gleichem Inhalt) werden durch 'tabelle1'.spalte1='tabelle2'.spalte2' verknüpft. Hinweis: Da die Spalte 'name' in der Tabelle 'mitarbeiter' und 'abteilungen' gleich heißen, muß bei Aufruf der Spalte der Tabellennamenname davor gestellt und mit einem neuen Spaltennamen versehen werden.

```
SELECT abteilung, mitarbeiter.name "Mitarbeitername", funktion, budget, leiter, gehaltsstufe  
FROM mitarbeiter, abteilungen WHERE mitarbeiter.abteilung=abteilungen.name;
```

- Durch ORDER BY 'spalte' ASC wird die Ausgabe aufsteigend nach der Spalte 'abteilung' sortiert (absteigend mit DSC) Hinweis: Die Sortierung bei Buchstaben erfolgt durch: A>B>a>b

```
SELECT * FROM aufgabe4a ORDER BY abteilung ASC;
```

- Ausgabe der Spalten 'mitarbeitername', 'funktion', 'budget', 'leiter' und 'gehaltsstufe', sortiert nach 'abteilung'.

b) Erstellen einer Liste aller Karten, die eine Kartenfläche haben die kleiner ist als die Karte von Saarbrücken (ohne die Fläche der Vergleichskarte zu ermitteln).

```
/*  
Aufgabe 4b */
```

```
DROP VIEW aufgabe4b;
```

←DROP VIEW löscht, falls vorhanden, die Sicht 'aufgabe4b'.

```
CREATE VIEW aufgabe4b AS
```

←CREATE VIEW 'aufgabe4b' AS kreiert die Sicht 'aufgabe4b'.

- Unterabfragen (sog. subqueries) können durch einen weiteren SELECT-Befehl in Klammern realisiert werden. Die Anzahl an Unterabfragen ist dabei unbegrenzt.

```
SELECT titel FROM karten WHERE breite*höhe<(SELECT breite*höhe FROM karten WHERE titel='Saarbrücken');
```

```
SELECT * FROM aufgabe4b;
```

- Ausgabe der Karten aus der Tabelle 'karten' (Spalte 'titel')

### Aufgabe 5:

a) Erstellen einer Liste mit der Anzahl der Touristik- und Panoramakarten, sowie deren durchschnittlichen Preis.

```
/*  
Aufgabe 5a */
```

```
DROP VIEW aufgabe5a;
```

←DROP VIEW löscht, falls vorhanden, die Sicht 'aufgabe5a'.

```
CREATE VIEW aufgabe5a AS
```

←CREATE VIEW 'aufgabe5a' AS kreiert die Sicht 'aufgabe5a'.

- Der arithmetische Befehl COUNT('spalte') zählt die Anzahl der Elemente in der entsprechenden Spalte. Hier werden von der Tabelle 'karten' mit der Bedingung, daß in der Spalte 'inhalt' die Daten ausgewählt werden, die entweder eine Panoramakarte oder eine Touristikkarte ist. Die ermittelte Anzahl an Karten wird in der neu benannten Spalte 'anzahl' ausgegeben. Der durchschnittliche Preis wird durch die arithmetische Funktionen ROUND und AVG (Berechnung des durchschnittlichen Preises), auf zwei Stellen hinter dem Komma gerundet, berechnet.

```
SELECT COUNT(*) "anzahl", ROUND(AVG(preis),2) "durchschnittlicher preis"  
FROM karten GROUP BY inhalt HAVING inhalt='Panorama' OR inhalt='Touristik';
```

```
SELECT * FROM aufgabe5a;
```

↑

- Ausgabe der Anzahl an Karten aus der Tabelle 'karten' und der durchschnittliche Preis dieser.

**b) Gruppierung der Karten nach Preisklassen durch Einfügen der Spalte 'preisklasse' in die Tabelle 'karten' mit einem Befehl (die Preisklassen sind dabei in 5.- DM - Schritten eingeteilt).**

```
/*  
Aufgabe 5b */
```

- Durch den Befehl ALTER TABLE 'tabelle' ADD ('spalte' Datentyp) wird eine neue Spalte (mit definition des Datentyps) an die Tabelle 'tabelle' angefügt.  
Hier wird der Tabelle 'karten' durch die Spalte 'preisklasse' mit zweistelliger Ziffer erweitert.

```
ALTER TABLE karten ADD (preisklasse NUMBER (2));
```

- In der neuen Spalte 'preisklasse' werden nun die Werte (hier: für die Gruppierung) berechnet. Der Befehl TRUNC schneidet bei der Berechnung die Nachkommastellen ab.  
Hinweis: Damit es keine nullte Klasse gibt wird die Spalte um eins erhöht.

```
UPDATE karten SET preisklasse=TRUNC (preis/5)+1;
```

- Berechnung der Anzahl der Karten pro Preisklasse durch COUNT und entsprechend durch die gruppierte Preisklassen der Tabelle 'karten'.  
Hinweis: Zur zusätzlichen Ausgabe der Bereiche der Preisklassen werden neue Spalten mit Hilfe der Spalte 'preisklasse' erstellt und ausgegeben.

```
SELECT preisklasse, preisklasse*5-5 "von", preisklasse*5 "bis unter DM",  
COUNT(preisklasse) "Anzahl Karten pro pk" FROM karten GROUP BY preisklasse ;
```

- Ausgabe der Preisklassen, deren Bereiche, sowie die Anzahl der Karten pro Preisklasse

**c) Siehe Aufgabe 3c)**

## Aufgabe 6:

**Generierung von Views der Tabellen ohne den Spalten nach eigener Wahl, mit entsprechender Abfrage nach jedem View.**

```
/*  
Aufgabe 6a */
```

```
DROP VIEW aufg6_1;
```

←DROP VIEW löscht, falls vorhanden, die Sicht 'aufgabe6\_1'.

```
CREATE VIEW aufg6_1 AS
```

←CREATE VIEW 'aufgabe6\_1' AS kreiert die Sicht 'aufgabe6\_1'.

• Auswahl der Spalten

↓

```
SELECT lfd, titel, modul, breite, höhe, zustand, ausgabe, inhalt, preis FROM karten;
```

```
SELECT lfd, titel, modul, breite, höhe, zustand, ausgabe, inhalt, preis FROM karten;
```

↑

• Ausgabe der Sicht

Die folgenden Tabellen werden auf der selben Art ausgegeben!

```
DROP VIEW aufg6_2;
```

```
CREATE VIEW aufg6_2 AS
```

```
SELECT name, budget, deckungsbeitrag FROM abteilungen;
```

```
SELECT name, budget, deckungsbeitrag FROM abteilungen;
```

```
DROP VIEW aufg6_3;
```

```
CREATE VIEW aufg6_3 AS
```

```
SELECT name, abteilung, funktion FROM mitarbeiter;
```

```
SELECT name, abteilung, funktion FROM mitarbeiter;
```

```
DROP VIEW aufg6_4;
```

```
CREATE VIEW aufg6_4 AS
```

```
SELECT region, karten, atlanten, sonstige FROM erloese;
```

```
SELECT region, karten, atlanten, sonstige FROM erloese;
```

## **Teil 6**

Indexverzeichnis



---

<b>A</b>		<b>I</b>	
ADD MONTHS.....	6	INSERT INTO.....	2
ALTER TABLE.....	7; 12	<b>N</b>	
ASC.....	10	NUMBER.....	2
AVG.....	11	<b>O</b>	
<b>B</b>		ORDER BY.....	10
BETWEEN...AND.....	5; 6	<b>R</b>	
<b>C</b>		ROUND.....	6; 8; 11
CHAR.....	2	<b>S</b>	
COUNT.....	11; 12	SELECT.....	5; 10
CREATE TABLE.....	2	SPOOL-Datei.....	1
<b>D</b>		START-Datei.....	1
DATE.....	2	SUM.....	8
DATENTYPEN.....	2	<b>T</b>	
DROP TABLE.....	2	TRUNC.....	12
DSC.....	10	<b>W</b>	
<b>G</b>		WHERE.....	7; 8; 10
GROUP BY.....	8	WHERE...AND.....	5
<b>H</b>			
HAVING.....	8; 9		