

Familienname, Vorname (bitte durch eine Leerspalte trennen)

[illegible]

Berufsnummer

IHK-Nummer

Prüflingsnummer

5	5
---	---

1	1
---	---

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	96	97	98	99	100
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----

--	--	--	--	--

Sp. 1-2

Sp. 3-6

Sp. 7-9

Sp. 10-14

## 1196

# 1

Fachinformatiker  
Fachinformatikerin  
Anwendungsentwicklung

# IHK

5 Handlungsschritte  
90 Minuten Prüfungszeit  
100 Punkte

## Bearbeitungshinweise

1. Der vorliegende Aufgabensatz besteht aus insgesamt 5 Handlungsschritten zu je 25 Punkten.

In der Prüfung zu bearbeiten sind 4 Handlungsschritte, die vom Prüfungsteilnehmer frei gewählt werden können.

Der nicht bearbeitete Handlungsschritt ist durch Streichung des Aufgabentextes im Aufgabensatz und unten mit dem Vermerk „Nicht bearbeiteter Handlungsschritt: Nr. ... „ an Stelle einer Lösungsniederschrift deutlich zu kennzeichnen. Erfolgt eine solche Kennzeichnung nicht oder nicht eindeutig, gilt der 5. Handlungsschritt als nicht bearbeitet.

2. Füllen Sie zuerst die **Kopfzeile** aus. Tragen Sie Ihre Familiennamen, Ihren Vornamen und Ihre Prüflings-Nr. in die oben stehenden Felder ein.
3. Lesen Sie bitte den **Text** der Aufgaben ganz durch, bevor Sie mit der Bearbeitung beginnen.
4. Halten Sie sich bei der Bearbeitung der Aufgaben genau an die **Vorgaben der Aufgabenstellung** zum Umfang der Lösung. Wenn z. B. vier Angaben gefordert werden und Sie sechs Angaben anführen, werden nur die ersten vier Angaben bewertet.
5. Tragen Sie die frei zu formulierenden **Antworten dieser offenen Aufgabenstellungen** in die dafür lt. Aufgabenstellung vorgesehenen Bereiche (Lösungszeilen, Formulare, Tabellen u. a.) des Arbeitsbogens ein.
6. Sofern nicht ausdrücklich ein Brief oder eine Formulierung in ganzen Sätzen gefordert werden, ist eine **stichwortartige Beantwortung** zulässig.
7. Verwenden Sie nur einen Kugelschreiber und schreiben Sie deutlich und gut lesbar. Ein nicht eindeutig zuzuordnendes oder **unleserliches Ergebnis** wird als **falsch** gewertet.
8. Zur Lösung der Rechenaufgaben darf ein nicht programmierter, netzunabhängiger **Taschenrechner** ohne Kommunikationsmöglichkeit mit Dritten verwendet werden.
9. Wenn Sie ein **gerundetes Ergebnis** eintragen und damit weiterrechnen müssen, rechnen Sie (auch im Taschenrechner) nur mit diesem gerundeten Ergebnis weiter.
10. Für **Nebenrechnungen/Hilfsaufzeichnungen** können Sie das im Aufgabensatz enthaltene Konzeptpapier verwenden. Dieses muss vor Bearbeitung der Aufgaben herausgetrennt werden. Bewertet werden jedoch nur Ihre Eintragungen im Aufgabensatz.

**Wird vom Korrektor ausgefüllt!**

## Bewertung

Für die Bewertung gilt die Vorgabe der Punkte in den Lösungshinweisen. Für den abgewählten Handlungsschritt ist anstatt der Punktzahl die Buchstabenkombination „AA“ in die Kästchen einzutragen.

Nicht bearbeiteter Handlungsschritt ist Nr.

Spalte 1 - 14  
S. o.

Punkte  
1. Handlungsschritt  
15 16

Punkte  
2. Handlungsschritt  
17 18

Punkte  
3. Handlungsschritt  
19 20

Punkte  
4. Handlungsschritt  
21 22

Punkte  
5. Handlungsschritt  
23 24

Gesamtpunktzahl  
26 27 28

Befreiungssatz-Datum

Prüfungszeit

25

Die entsprechende Ziffer (1, 2 oder 3) finden Sie in der Abfrage nach der Prüfungszeit im Anschluss an die letzte Aufgabe.

Unterschrift

Gemeinsame Prüfungsaufgaben der Industrie- und Handelskammern. Dieser Aufgabensatz wurde von einem überregionalen Ausschuss, der entsprechend § 40 Berufsbildungsgesetz zusammengesetzt ist, beschlossen.  
Die Vervielfältigung, Verbreitung und öffentliche Wiedergabe der Prüfungsaufgaben und Lösungen ist nicht gestattet. Zuwiderhandlungen werden zivil- und strafrechtlich (§§ 97 ff., 106 ff. UrhG) verfolgt. – © ZPA Nord-West 2018 – Alle Rechte vorbehalten!

**Die Handlungsschritte 1 bis 5 beziehen sich auf die folgende Ausgangssituation:**

Sie sind Mitarbeiter/-in der eCo GmbH, Astadt.

Die eCo GmbH ist ein Softwarehaus.

Die eCo GmbH wurde von der Pharmalog GmbH, einem Großhändler für Arzneimittel, mit der Software-Ausstattung eines neuen Lagers beauftragt.

Sie arbeiten in diesem Projekt mit.

Sie sollen vier Aufgaben aus den folgenden fünf Bereichen erledigen:

1. Projektplanung
2. Objektorientierte Programmierung
3. Prozedurale Programmierung
4. Datenbankmodellierung
5. SQL

**1. Handlungsschritt (25 Punkte)**

Die eCo GmbH plant das Teilprojekt „LAN“ für die Pharmalog GmbH.

a) Das Teilprojekt „LAN“ wurde wie folgt geplant.

Vorgang	Beschreibung	Dauer	Vorgänger
A	Planung	4	-
B	Beschaffung Hardware	5	A
C	Beschaffung Standard-Software	2	A
D	Erstellung Individual-Software	10	A
E	Installation Netzwerkkomponenten	7	B
F	Installation Software und Computer	4	C, D
G	Tests und Übergabe	7	E, F

Erstellen Sie auf Seite 3 anhand der Vorgangsliste einen Netzplan und kennzeichnen Sie den kritischen Pfad.

16 Punkte

b) Die Pharmalog GmbH legt besonders großen Wert auf die Qualität der von ihr entwickelten Software.

ba) Die Software soll folgende Qualitätsmerkmale besitzen:

– Portability – Maintainability – Efficiency – Reliability – Functionality – Usability

Erläutern Sie drei dieser Qualitätsmerkmale.

6 Punkte

1. Qualitätsmerkmal:

---



---



---

2. Qualitätsmerkmal:

---



---



---

3. Qualitätsmerkmal:

---



---



---

bb) Nennen Sie drei Maßnahmen zur Qualitätssicherung.

3 Punkte

---



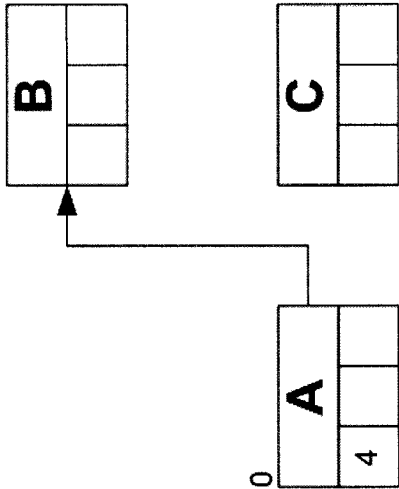
---



---



---



Vorgangsliste

Vorgang	Beschreibung	Dauer	Vorgänger
A	Planung	4	-
B	Beschaffung Hardware	5	A
C	Beschaffung Standard-Software	2	A
D	Erstellung Individual-Software	10	A
E	Installation Netzwerkkomponenten	7	B
F	Installation Software und Computer	4	C, D
G	Tests und Übergabe	7	E, F

FAZ	FEZ
Vorgang	
Dauer	GP
SAZ	FP
SEZ	

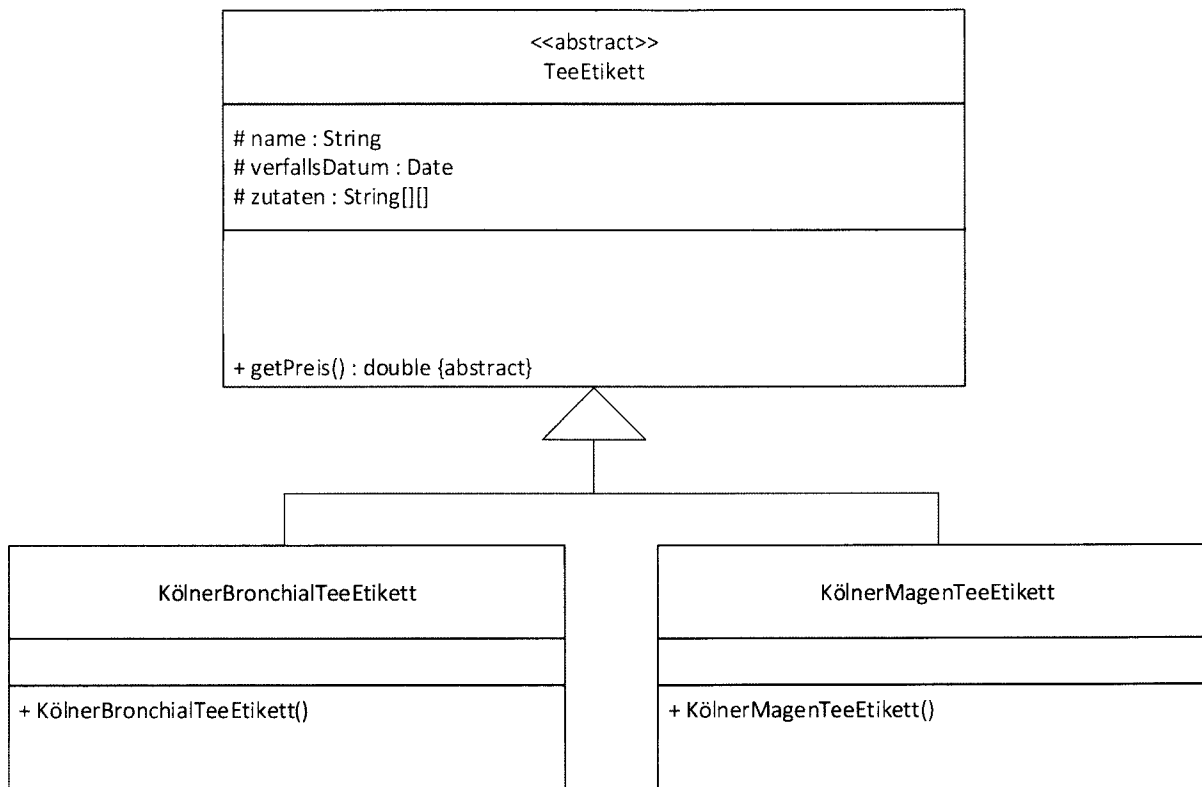
Vorgang	Vorgangs-ID (A, B, C, ...)
Dauer	Dauer in Arbeitstagen
FAZ	Frühester Anfangszeitpunkt
FEZ	Frühester Endzeitpunkt
SAZ	Spätester Anfangszeitpunkt
SEZ	Spätester Endzeitpunkt
GP	Gesamtpuffer, GP = SAZ – FAZ oder GP = SEZ – FEZ
FP	Freier Puffer, FP = FAZ des Nachfolgers – FEZ des Vorgangs

## 2. Handlungsschritt (25 Punkte)

Korrekturrand

Die eCo GmbH soll für die Pharmalog GmbH ein Programm erstellen, mit dem Etiketten für Tee-Verpackungen gedruckt werden können.

- a) Im Lagerbereich werden auch Teemischungen hergestellt, die etikettiert werden müssen.  
Für die objektorientierte Drucksoftware liegt ein grober, noch unvollständiger Entwurf in Form eines Klassendiagramms bereits vor:



Die Instanzvariablen *name* und *zutaten* werden in den jeweiligen Konstruktoren initialisiert. Alle TeeEtikett-Klassen benötigen folgende öffentliche Methoden:

Funktion	Beschreibung
<i>berechneVerfallsDatum</i>	Legt für alle Teesorten in gleicher Weise die Instanzvariable <i>verfallsDatum</i> fest und gibt nichts zurück.
<i>druckeEtikett</i>	Erledigt für alle Teeetiketten in gleicher Weise den Ausdruck und gibt nichts zurück.
<i>getPreis</i>	Gibt den Preis der jeweiligen Teemischung als Dezimalzahl zurück.

aa) Ergänzen Sie im Klassendiagramm die fehlenden UML-Methodenbeschreibungen. 4 Punkte

ab) Erläutern Sie die Art der Beziehung zwischen den Klassen. 2 Punkte

ac) Erläutern Sie die Bedeutung der *abstract*-Eigenschaft in Zusammenhang mit der Klasse *TeeEtikett* und

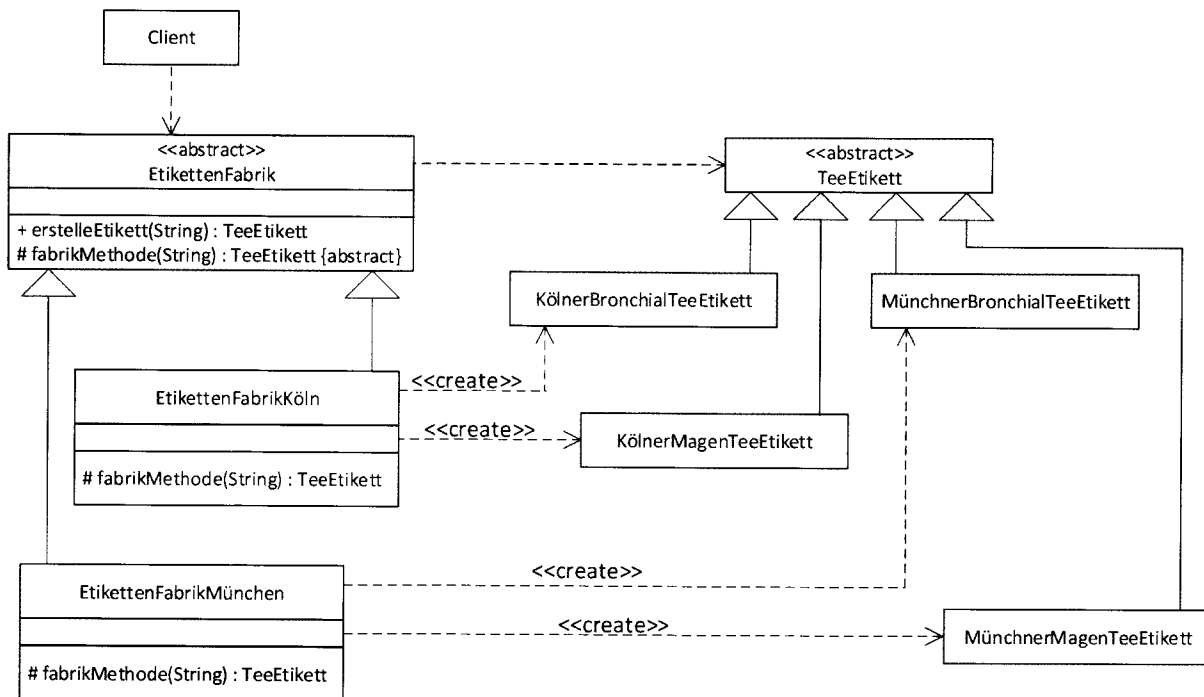
der Methode *getPreis* der *TeeEtikett*-Klasse.

2 Punkte

- b) Das Angebot an Teemischungen und auch deren Zusammensetzung wird laufend verändert. Deswegen soll die Verwendung von *TeeEtikett*-Objekten von ihrer Implementierung unabhängig sein.

Korrekturrand

Ein Kollege schlägt folgenden Entwurf vor:



Hinweis: Der Bezeichner für die String-Übergabeparameter soll *typ* sein.  
String-Übergabeparameter = Bezeichnung für die Etiketten, z. B. „Bronchial“

- ba) Implementieren Sie beispielhaft in Pseudocode die Methode *fabrikMethode* der Klasse *EtikettenFabrikKöln* für die übergebene Zeichenkette „Bronchial“.

3 Punkte

---

---

---

---

---

---

---

---

- bb) Implementieren Sie in Pseudocode die Methode *erstelleEtikett*.

Die Methode *erstelleEtikett* soll mithilfe der Methode *fabrikMethode* ein Etikett erstellen und für dieses Etikett die Methode *berechneVerfallsDatum* der Klasse *TeeEtikett* aufrufen.

4 Punkte

---

---

---

---

---

---

---

---

## Fortsetzung 2. Handlungsschritt

Korrekturrand

- c) Die Fächer in den Regalen des Lagers der Pharmalog GmbH können eine bestimmte Anzahl (max) an Gegenständen aufnehmen. Daher kann jedes Fach einen der folgenden Zustände annehmen:

leer: Das Fach enthält keine Gegenstände. Dies ist direkt nach der Einrichtung des Faches oder nach einer vollständigen Leerung der Fall.

teilbelegt: Das Fach enthält Gegenstände, kann aber weitere (bis max Gegenstände) aufnehmen.

voll: Das Fach ist vollständig belegt.

Ein Fach wird beim Start eingerichtet und am Ende aufgelöst.

Der Übergang von einem Zustand zu einem anderen wird durch die Ereignisse *hineinlegen* bzw. *entnehmen* ausgelöst.



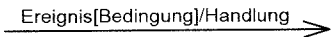
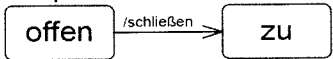
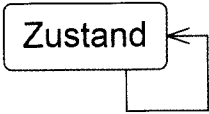

Dabei ist ein Zustandsübergang durch das Ereignis *hineinlegen* vom Zustand *leer* in den Zustand *teilbelegt* oder *voll* möglich.

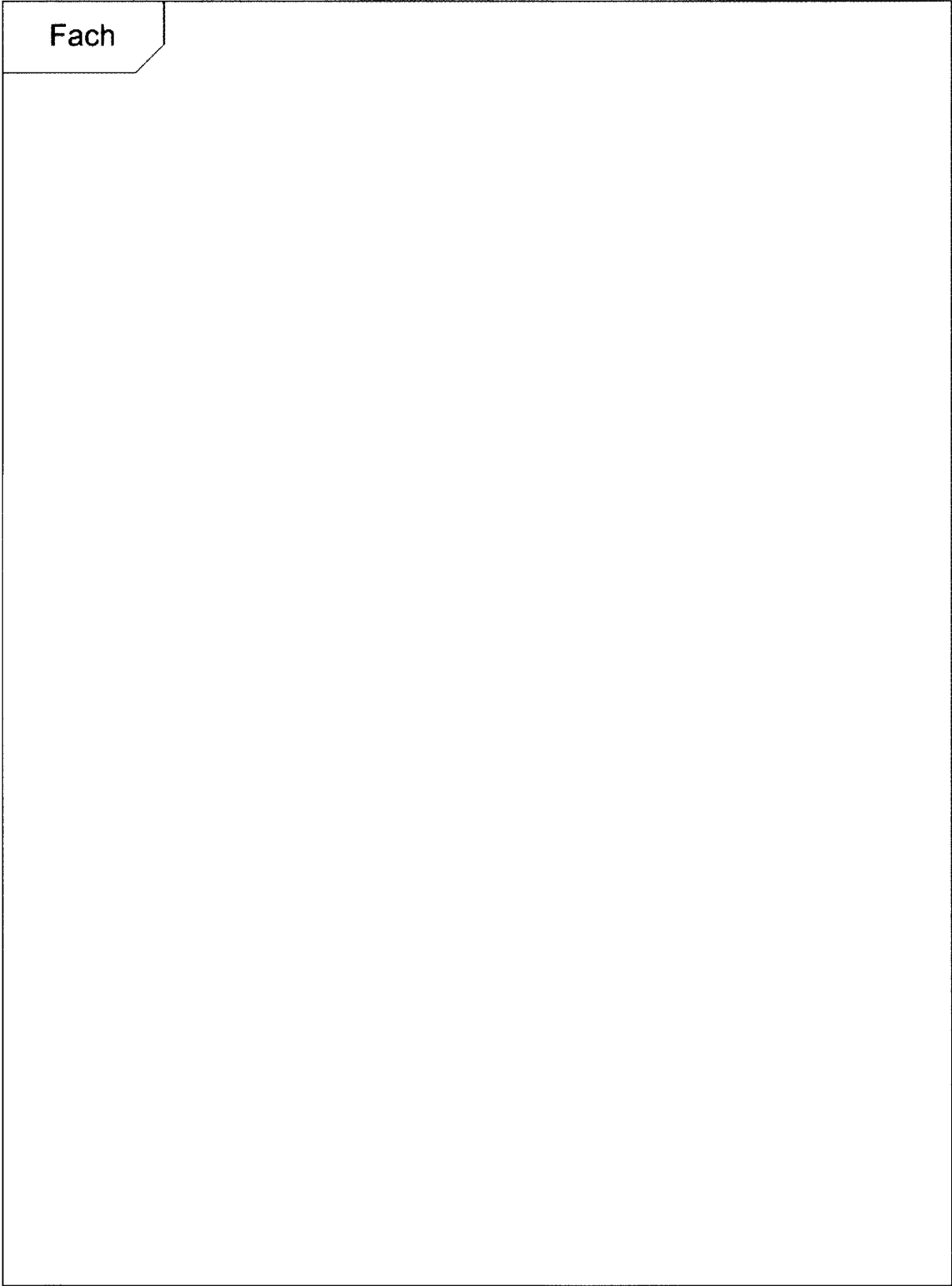
Durch das Ereignis *entnehmen* ist ein Zustandsübergang vom Zustand *voll* oder *teilbelegt* in den Zustand *teilbelegt* oder *leer* möglich.

Durch die Ereignisse *hineinlegen* oder *entnehmen* kann der Zustand *teilbelegt* erhalten bleiben.

Erstellen Sie für das beschriebene Szenario ein UML-Zustandsdiagramm für ein Fach des Lagers.

10 Punkte

	Anfangszustand
	Zustand, den ein Element einnehmen kann. z. B. Element = Fenster; Zustände: offen, geschlossen
 <b>Beispiel</b> 	Zustandsübergang (Transition) von einem Quellzustand zu einem Zielzustand. In der Beschriftung kann Folgendes angegeben werden: <ul style="list-style-type: none"> <li>- Ein Ereignis (trigger), welches den Zustandsübergang auslöst</li> <li>- Eine Bedingung (guard), welche beim Zustandsübergang erfüllt sein muss</li> <li>- Eine Handlung (action), welche den Zustandsübergang bewirkt</li> </ul>
 Ereignis[Bedingung]/Handlung	Selbstaufruf
	Endzustand



### 3. Handlungsschritt (25 Punkte)

Korrekturrand

Die Pharmalog GmbH setzt Picker ein. Die Picker sind Geräte, die gemäß einer Bestellliste Fächer im Lager anfahren und die entsprechende Ware entnehmen.

Die Bestellliste liegt in einem zweidimensionalen Array *liste* sortiert vor. Jede Zeile dieses Arrays enthält eine Ware-ID (Spalte 0), eine Regalnummer (Spalte 1), eine Ebenennummer (Spalte 2) und eine Fachnummer (Spalte 3).

Array *liste* (Beispiel)

	Ware-ID	Position		
	Regal	Ebene	Fach	
	[0]	[1]	[2]	[3]
[0]	434	1	1	1
[1]	434	1	1	4
[2]	189	3	3	12
[3]	201	4	2	8
...	...	...	...	...

Der Picker soll zu den im Array angegebenen Positionen fahren. Bevor er aus einem Fach etwas entnimmt, soll er prüfen, ob die ID der Ware im Fach mit der Ware-ID im Array übereinstimmt. Bei einer Übereinstimmung soll der Picker die Ware entnehmen, andernfalls soll die aktuelle Zeile des Array *liste* in ein zweidimensionales Array *fehler* übertragen werden. Das Array *fehler* steht bereits zur Verfügung.

Folgende Funktionen sollen verwendet werden:

Funktion	Beschreibung
fahreRegalAn(regalNr : int)	Fährt den Picker zum Regal <i>regalNr</i> , Ebene 1, Fach 1
fahreEbeneAn(ebeneNr : int)	Fährt den Picker im aktuellen Regal zur Ebene <i>ebeneNr</i> , Fach 1
fahreFachAn(fachNr : int)	Fährt den Picker im aktuellen Regal, in der aktuellen Ebene zum Fach <i>fachNr</i>
pruefeWare(ware_id : int) : boolean	Prüft, ob in dem aktuellen Fach die Ware mit der Id <i>ware_id</i> vorhanden ist
entnehmeWare(regalNr : int, ebeneNr : int, fachNr : int)	Entnimmt die Ware im angegebenen Fach und meldet die Entnahme dem System
kopiereZeile(liste : zweidim. Array, fehler : zweidim. Array, zeileL : int, zeileF : int)	Kopiert die Zeile <i>zeileL</i> aus dem Array <i>liste</i> in die Zeile <i>zeileF</i> des Arrays <i>fehler</i>

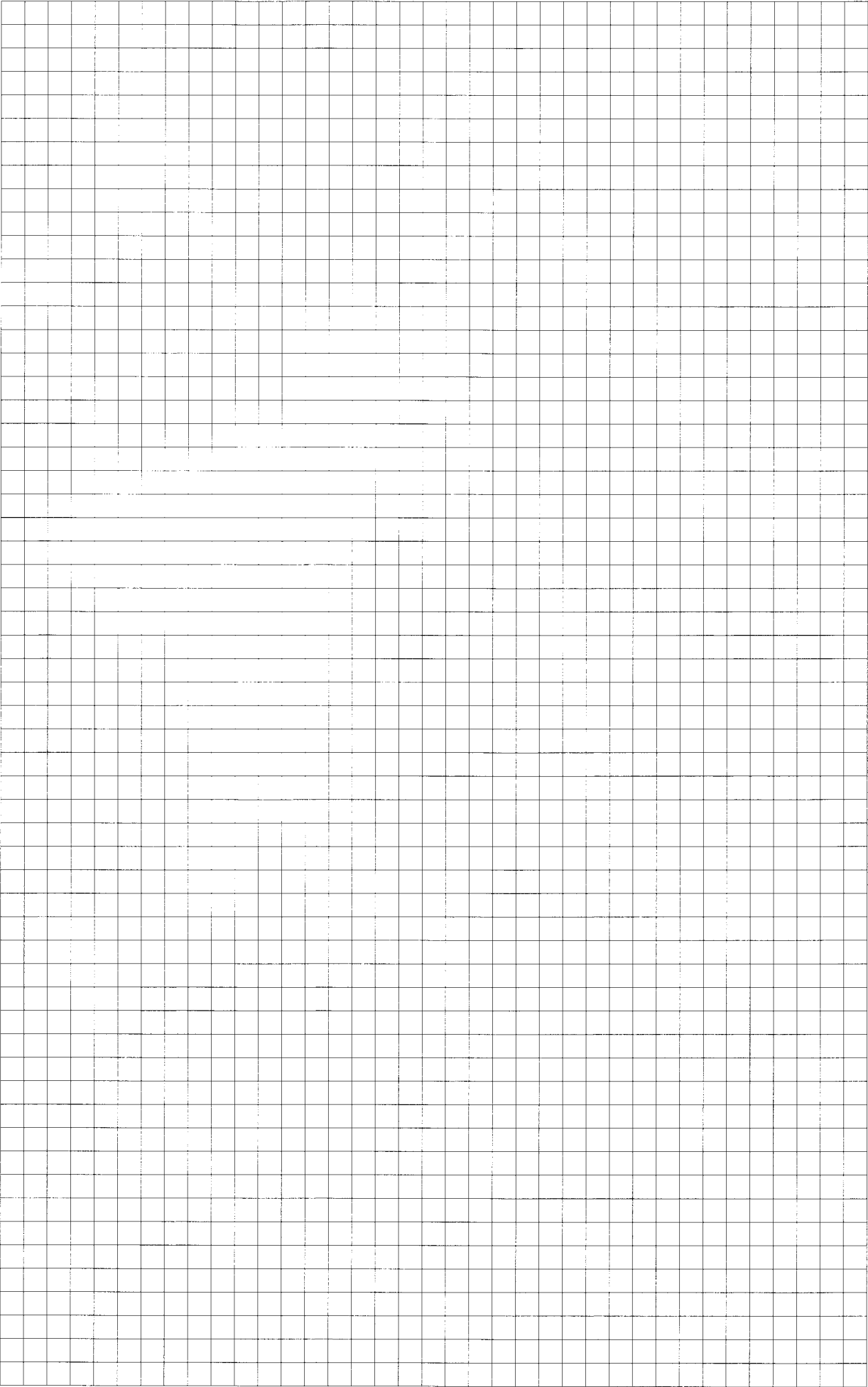
Erstellen Sie eine Prozedur *entnehmeWaren(liste : zweidimensionales Array vom Typ int)*, mit welcher der Picker den Anforderungen entsprechend gesteuert werden kann. Verwenden Sie dazu die angegebenen Funktionen.

Stellen Sie den Algorithmus in Pseudocode, in einem Struktogramm oder einem Programmablaufplan dar.



entnehmeWaren (liste : zweidimensionales Array vom Typ int)

Korrekturrand



#### 4. Handlungsschritt (25 Punkte)

Korrekturrand

Die eCo GmbH soll für die Pharmalog GmbH eine Datenbank erstellen.

Der Sachverhalt wird wie folgt beschrieben:

Ein Einkäufer der Pharmalog GmbH führt mehrere Bestellungen aus.

Mit jeder Bestellung werden eine oder mehrere Waren zu unterschiedlichen Stückzahlen bestellt.

Eine Ware kann jeweils nur bei einem Hersteller bezogen werden.

Jede Ware wird einer Warengruppe zugeordnet.

An einem Lagerort können mehrere Waren gelagert werden.

- a) Erstellen Sie auf der gegenüberliegenden Seite zum dargestellten Sachverhalt das entsprechende Entity-Relationship-Modell (ER-Modell).

Hinweis: Das Attribut *Stückzahl* soll im ER-Modell angegeben werden.

18 Punkte

- b) Die Tabellen *Hersteller* und *Ware* der Datenbank wurden erstellt und mit Testdaten gefüllt.

Diese Testdaten sollen wieder gelöscht werden. Die Tabellen sind über das Attribut *Hersteller\_ID* miteinander verknüpft.

Erläutern Sie, wie diese Löschung durchgeführt werden muss.

5 Punkte

---

---

---

---

---

- c) Erläutern Sie, wie die Beziehung zwischen Ware und Bestellung im relationalen Datenbankmodell abgebildet wird.

2 Punkte

---

---

---

## Korrekturrand

```
graph LR
    Kunde --> Rechnung
    Rechnung --> Artikel
    Artikel --> Hersteller
    Rechnung --> Position
    Artikel --> Position
```

The diagram illustrates the following tables and their attributes:

- Kunde**: Kunde\_ID (PK), Firma, Strasse, Plz, Ort
- Rechnung**: Rechnung\_ID (PK), Kunde\_ID (FK), RechnungsDatum
- Artikel**: Artikel\_ID (PK), Hersteller\_ID (FK), Bezeichnung, Listenpreis
- Position**: Rechnung\_ID (FK), Artikel\_ID (FK), Menge, Verkaufs\_Einzelpreis
- Hersteller**: Hersteller\_ID (PK), Firma

Relationships are indicated by lines connecting the tables:

- Kunde** to **Rechnung**: 1:1 relationship on Kunde\_ID.
- Rechnung** to **Artikel**: 1:1 relationship on Artikel\_ID.
- Artikel** to **Hersteller**: 1:1 relationship on Hersteller\_ID.
- Rechnung** to **Position**: 1:1 relationship on Rechnung\_ID.
- Artikel** to **Position**: 1:1 relationship on Artikel\_ID.

- aa) Erstellen einer Liste aller Kunden mit Anzahl der Rechnungen im Jahr 2015.  
Ausgegeben werden sollen: Kunde\_ID, Firma und Anzahl.

7 Punkte

762762	ABC-Gesund AG	54
872346	Zeppelin-Pharma GmbH	54
138787	FeelGood-Partner OHG	49
...		
122636	Loser&Vaul GmbH	0

- ab) Erhöhen der Listenpreise aller Artikel vom Hersteller BigPill AG um 4,5 %.

4 Punkte

- ac) Erstellen einer Liste aller Kunden mit dem Umsatz je bezogenem Artikel.  
Ausgegeben werden sollen: Kunde\_ID und Firma des Kunden sowie Bezeichnung und Umsatz der Artikel.

8 Punkte

Korrekturrand

Beispiel

762762	ABC-Gesund AG	Aspiran forte	28.600,00
762762	ABC-Gesund AG	Cardioshock 2.0	17.300,00
...			
872346	Zeppelin-Pharma GmbH	Cardioshock 2.0	38.500,00
...			
138787	FeelGood-Partner OHG	Katerstop	19.800,00
...			

- b) Die Datenbank soll erweitert werden. Dazu sollen Sie für folgende Aufgaben SQL-Anweisungen erstellen.

- ba) Erstellen einer Tabelle Artikelgruppe mit einem ganzzahligen Primärschlüsselattribut und der Artikelgruppenbezeichnung.

3 Punkte

- bb) Einbinden der neuen Tabelle Artikelgruppe in die Datenbank.

3 Punkte

## PRÜFUNGSZEIT – NICHT BESTANDTEIL DER PRÜFUNG!

Wie beurteilen Sie nach der Bearbeitung der Aufgaben die zur Verfügung stehende Prüfungszeit?

- ☐ 1 Sie hätte kürzer sein können. ☐ 2 Sie war angemessen. ☐ 3 Sie hätte länger sein müssen.



**Dieses Blatt kann an der Perforation aus dem Aufgabensatz herausgetrennt werden!**

SQL-Syntax (Auszug)

Syntax	Beschreibung
<b>Tabelle</b>	
CREATE TABLE Tabellename( Spaltenname < DATENTYP >, Primärschlüssel, Fremdschlüssel)	Erzeugt eine neue leere Tabelle mit der beschriebenen Struktur
ALTER TABLE Tabellename ADD COLUMN Spaltenname Datentyp DROP COLUMN Spaltenname  ADD FOREIGN KEY(Spaltenname) REFERENCES Tabellename( Primärschlüsselspaltenname )	Änderungen in einer Tabelle: Hinzufügen einer Spalte Entfernen einer Spalte  Definiert eine Spalte als Fremdschlüssel
CHARACTER	Textdatentyp
DECIMAL	Numerischer Datentyp (Festkommazahl)
DOUBLE	Numerischer Datentyp (Doppelte Präzision)
INTEGER	Numerischer Datentyp (Ganzzahl)
DATE	Datum (Format DD.MM.YYYY)
PRIMARY KEY (Spaltenname)	Erstellung eines Primärschlüssels
FOREIGN KEY (Spaltenname) REFERENCES Tabellename( Primärschlüsselspaltenname )	Erstellung einer Fremdschlüssel-Beziehung
DROP TABLE Tabellename	Löscht eine Tabelle
<b>Befehle, Klauseln, Attribute</b>	
SELECT *   Spaltenname1 [, Spaltenname2, ...]	Wählt die Spalten einer oder mehrerer Tabellen, deren Inhalte in die Liste aufgenommen werden sollen; alle Spalten (*) oder die namentlich aufgeführten
FROM	Name der Tabelle oder Namen der Tabellen, aus denen die Daten der Ausgabe stammen sollen
SELECT ... (SELECT ... FROM ... WHERE ...) AS xyz FROM ... WHERE ...	Unterabfrage, die in eine äußere SELECT-Anweisung geschachtelt ist. Das Ergebnis der Unterabfrage wird im Spaltenausdruck (z. B. hier: xyz) ausgegeben.
INNER JOIN	Liefert nur die Datensätze zweier Tabellen, die gleiche Datenwerte enthalten
LEFT JOIN / Left OUTER JOIN	Liefert von der erstgenannten (linken) Tabelle alle Datensätze und von der zweiten Tabelle jene, deren Datenwerte mit denen der ersten Tabelle übereinstimmen
RIGHT JOIN / RIGHT OUTER JOIN	Liefert von der zweiten (rechten) Tabelle alle Datensätze und von der ersten Tabelle jene, deren Datenwerte mit denen der zweiten Tabelle übereinstimmen
FULL JOIN	Liefert aus beiden Tabellen jeweils alle Datensätze
WHERE	Bedingung, nach der Datensätze ausgewählt werden sollen
WHERE EXISTS ( subquery ) WHERE NOT EXISTS ( subquery )	Die Bedingungen EXISTS prüft, ob die Suchbedingung einer Unterabfrage mindestens eine Zeile zurückliefert. NOT EXIST negiert die Bedingung.
GROUP BY Spaltenname1 [,Spaltenname2, ...]	Gruppierung (Aggregation) nach Inhalt des genannten Feldes
ORDER BY Spaltenname1 [,Spaltenname2, ...] ASC   DESC	Sortierung nach Inhalt des genannten Feldes oder der genannten Felder ASC: aufsteigend; DESC: absteigend

Fortsetzung ->

Syntax	Beschreibung
<b>Datenmanipulation</b>	
DELETE FROM Tabellenname	Löschen von Datensätzen in der genannten Tabelle
UPDATE Tabellenname SET	Aktualisiert Daten in Feldern einer Tabelle
INSERT INTO Tabellenname VALUES (Wert für Spalte 1 [, Wert für Spalte 2, ...]) oder SELECT ... FROM ... WHERE	Fügt Datensätze in die genannte Tabelle, die entweder mit festen Werten belegt oder Ergebnis eines SELECT-Befehls sind
<b>Aggregatfunktionen</b>	
AVG(Spaltenname)	Ermittelt das arithmetische Mittel aller Werte im angegebenen Feld
COUNT(Spaltenname   *)	Ermittelt die Anzahl der Datensätze mit Nicht-NULL-Werten im angegebenen Feld oder alle Datensätze der Tabelle (dann mit Operator *)
SUM(Spaltenname   Formel)	Ermittelt die Summe aller Werte im angegebenen Feld oder der Formelergebnisse
MIN(Spaltenname   Formel)	Ermittelt den kleinsten aller Werte im angegebenen Feld
MAX (Spaltenname   Formel)	Ermittelt den größten aller Werte im angegebenen Feld
<b>Funktionen</b>	
LEFT(Zeichenkette, Anzahlzeichen)	Liefert <i>Anzahlzeichen</i> der Zeichenkette von links.
RIGHT(Zeichenkette, Anzahlzeichen)	Liefert <i>Anzahlzeichen</i> der Zeichenkette von rechts.
CURRENT	Liefert das aktuelle Datum mit der aktuellen Uhrzeit
CONVERT(time,[DatumZeit])	Liefert die Uhrzeit aus einer DatumZeit-Angabe
DATE(Wert)	Wandelt einen Wert in ein Datum um
DAY(Datum)	Liefert den Tag des Monats aus dem angegebenen Datum
MONTH(Datum)	Liefert den Monat aus dem angegebenen Datum
TODAY	Liefert das aktuelle Datum
WEEKDAY(Datum)	Liefert den Tag der Woche aus dem angegebenen Datum
YEAR(Datum)	Liefert das Jahr aus dem angegebenen Datum
DATEADD(Datumsteil, Intervall, Datum)	Fügt einem Datum ein Intervall (ausgedrückt in den unter Datumsteil angegebenen Einheiten) hinzu
DATEDIFF(Datumsteil, Anfangsdatum, Enddatum) Datumsteile: DAY, MONTH, YEAR	Liefert Enddatum-Startdatum (ausgedrückt in den unter Datumsteil angegebenen Einheiten)
<b>Operatoren</b>	
AND	Logisches UND
LIKE	Überprüfung von Textattributen auf Gleichheit, Verwendung von Platzhaltern möglich.
NOT	Logische Negation
OR	Logisches ODER
=	Test auf Gleichheit
>, >=, <, <=, < >	Test auf Ungleichheit
*	Multiplikation
/	Division
+	Addition, positives Vorzeichen
-	Subtraktion, negatives Vorzeichen

Stand 2017-03-18

S