

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

Sie sind Mitarbeiter/-in der Plan-B GmbH, Astadt.

Die Plan-B GmbH wurde vom Baumarkt ÖkoSpan KG mit der Restrukturierung der Informationstechnik beauftragt.

Sie arbeiten in diesem Projekt mit.

Sie sollen vier der folgenden fünf Aufgaben erledigen:

1. Ein Projekt planen
2. Software entwickeln
3. Einen Algorithmus erstellen
4. Eine Datenbank entwickeln
5. SQL-Anweisungen erstellen

1. Handlungsschritt (25 Punkte)

Sie arbeiten für die Plan-B GmbH, einem IT-Dienstleister für mittelständische Betriebe.

a) Der Plan-B GmbH liegt die Anfrage der ÖkoSpan KG für Softwareentwicklung vor.

Beschreiben Sie in folgender Tabelle Lastenheft und Pflichtenheft.

6 Punkte

	Lastenheft	Pflichtenheft
Verfasser		
Verwendung		
Inhalt		

b) Für das Teilprojekt „Zeiterfassung“ wurden folgende Vorgänge geplant, die von den genannten Mitarbeitern erledigt werden sollen.

Vorgang	Beschreibung	Dauer*	Vorgänger	Mitarbeiter
A	Planung	3	—	Dr. Huber, Fischer, Schneider
B	Softwareentwicklung	7	A	Schneider, Müller
C	Datenbankentwicklung	4	A	Kramer
D	Testphase	1	B, C	Fischer, Schneider
E	Installation, Integration	2	E	Müller, Fischer
F	Übergabe, Abnahme	1	F	Dr. Huber, Fischer, Schneider

* Dauer bei Einsatz der genannten Mitarbeiter

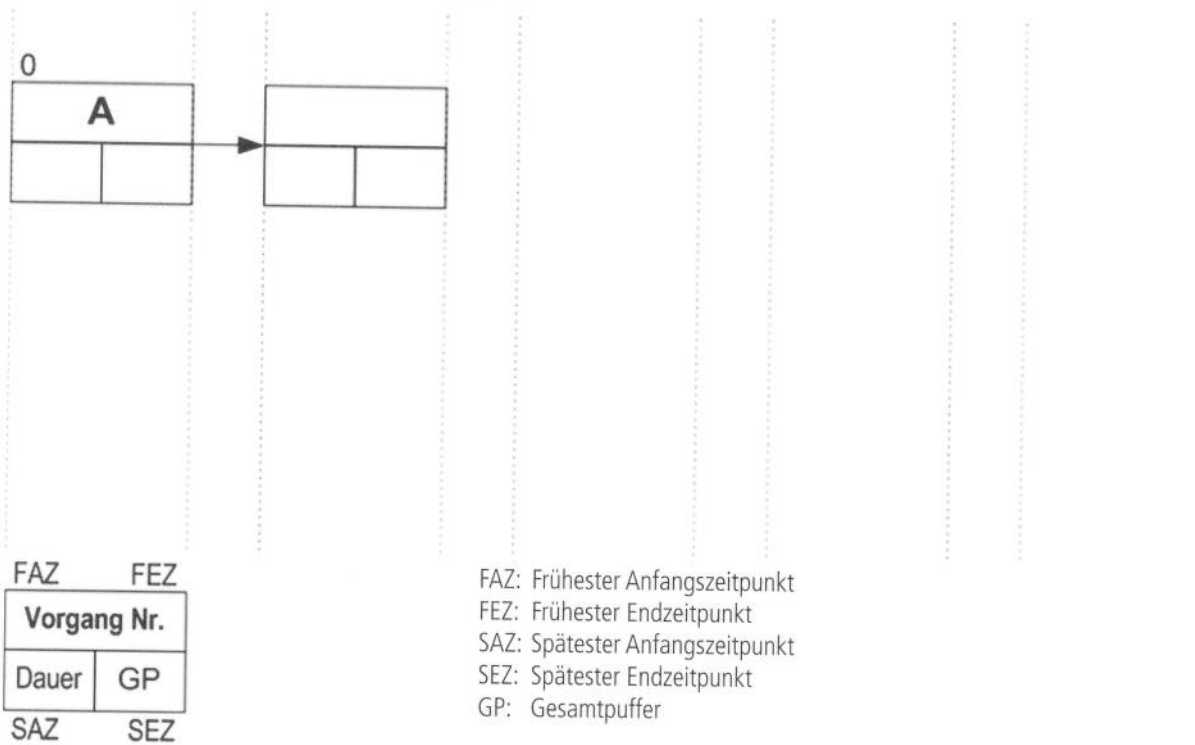
Die Übergabe des Teilprojekts soll spätestens am Freitag, 20.12.2013, erfolgen. Samstags und sonntags wird nicht gearbeitet.

ba) Erstellen Sie anhand der Vorgangsliste den Netzplan für dieses Teilprojekt und markieren Sie den kritischen Pfad.

Ergänzen Sie dazu den Netzplan auf der folgenden Seite.

8 Punkte

Netzplan für das Teilprojekt „Zeiterfassung“



bb) Vervollständigen Sie für dieses Teilprojekt den Personaleinsatzplan.

- Markieren Sie die Arbeitstage jeweils mit dem Kennbuchstaben des für diesen Tag geplanten Vorgangs (z. B. mit A für Planung).
- Lassen Sie die Vorgänge jeweils am frühest möglichen Tag beginnen.
- An den im Personaleinsatzplan geschwärzten Tagen sind die Mitarbeiter bereits für andere Arbeiten verplant. 6 Punkte

Personaleinsatzplan

	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr
Name	18	19	20	21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Dr. Huber																																	
Fischer																																	
Kramer																																	
Müller																																	
Schneider																																	

bc) Erstellen Sie anhand des Personaleinsatzplans den Projektplan für dieses Teilprojekt im vorbereiteten Gantt-Diagramm. Markieren Sie die Tage der Vorgänge jeweils mit X.

5 Punkte

Projektplan

	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr	Sa	So	Mo	Di	Mi	Do	Fr
Vorgang	18	19	20	21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
A Planung																																	
B SW-Entw.																																	
C DB-Entw.																																	
D Test																																	
E Inst./Int.																																	
F Übergabe																																	

2. Handlungsschritt (25 Punkte)

Korrekturrand

Die ÖkoSpan KG möchte ihren Mitarbeitern die Möglichkeit geben, jederzeit eine aktuelle Auswertung ihrer erfassten Arbeitszeiten eines Monats zu erhalten.

Angaben zur Zeiterfassung:

– Für jeden Tag werden maximal zwei Zeiten erfasst, Kommen- und Gehenzzeit. (Pausen werden nicht berücksichtigt.)

Der Report, der alle Buchungen eines Mitarbeiters für einen Monat anzeigt, soll wie folgt aufgebaut werden (siehe auch Beispiel).

- Liegen für einen Tag die Kommen- und Gehenzbuchungen vor, werden diese Zeiten und die berechnete Anwesenheitszeit in Stunden und Minuten angegeben.
- Liegt für einen Tag nur eine Zeitbuchung vor, ist diese Zeit als Kommenzeit, die Anwesenheitszeit 00:00 und der Text „Buchung fehlt“ auszugeben.
- Liegt für einen Tag keine Zeitbuchung vor, ist die Anwesenheitszeit 00:00 und der Text „nicht anwesend“ auszugeben.
- Zum Ende der Liste ist die Summe der Anwesenheitszeiten auszugeben.

Die Kommen- und Gehenzzeiten eines Mitarbeiters für einen Monat liegen in dem zweidimensionalen Array `zeiten` in Minuten vor (siehe Beispiel).

Erläuterung der Kommen- und Gehenzzeiten am Beispiel des 2. Mai 2013:

480 Minuten entsprechen 08:00 Uhr ($8 * 60$); 1.040 Minuten entsprechen 17:20 Uhr ($17 * 60 + 20$)

Array <code>zeiten</code>		Report				
Tag	Minuten	Mitarbeiter: 12345 Mai 2013				
2	480	Tag	Kommen	Gehen	Anwesenheit	Bemerkung
2	1.040	=====				
3	470	1			00:00	nicht anwesend
6	480	2	08:00	17:20	09:20	
6	960	3	07:50		00:00	eine Buchung fehlt
7	990	4			00:00	nicht anwesend
8	480	5			00:00	nicht anwesend
8	960	6	08:00	16:00	08:00	
		7	16:30		00:00	eine Buchung fehlt
		8	08:00	16:00	08:00	
30	990	30	16:60		00:00	eine Buchung fehlt
		31			00:00	nicht anwesend
		=====				
		Summe Anwesenheit: 43:10				

Folgende Funktionen sind bereits implementiert:

<code>tagImMonat(monat : int, jahr : int) : int</code>	Ermittelt die Anzahl der Tage für den übergebenen Monat eines Jahres
<code>schreibeKopfzeile(persnr : int, jahr : int, monat : int)</code>	Gibt die Kopfzeilen der Liste aus
<code>schreibeZeile(tag : int, min1 : int, min2 : int, anwesenheitTag : int, bemerkung : String)</code>	Gibt die Zeile für einen Tag aus; der Funktion sind zu übergeben: - die Buchungen in Minuten, für fehlen Buchungen der Wert -1 - die Dauer der Anwesenheit in Minuten; liegt nur eine oder keine Buchung vor, ist der Wert 0 zu übergeben - liegt keine Buchung vor, die Bemerkung „nicht anwesend“, bei einer Buchung „eine Buchung fehlt“ Die in Minuten übergebenen Zeiten werden von der Funktion in Stunden:Minuten ausgegeben, fehlende Buchungen als 00:00
<code>schreibeFusszeile(anwesenheitMonat : int)</code>	Gibt die Fußzeile aus. Die Monatsanwesenheit wird der Funktion in Minuten übergeben und von ihr in Stunden:Minuten ausgegeben.

erzeugeListe(persnr: int, zeiten: zweidim Tabelle vom Typ int, jahr : int, monat : int)

Korrekturrand

1. Modul 1.1
2. Modul 1.2
3. Modul 1.3
4. Modul 1.4
5. Modul 1.5
6. Modul 1.6
7. Modul 1.7
8. Modul 1.8
9. Modul 1.9
10. Modul 1.10
11. Modul 1.11
12. Modul 1.12
13. Modul 1.13
14. Modul 1.14
15. Modul 1.15
16. Modul 1.16
17. Modul 1.17
18. Modul 1.18
19. Modul 1.19
20. Modul 1.20
21. Modul 1.21
22. Modul 1.22
23. Modul 1.23
24. Modul 1.24
25. Modul 1.25
26. Modul 1.26
27. Modul 1.27
28. Modul 1.28
29. Modul 1.29
30. Modul 1.30
31. Modul 1.31
32. Modul 1.32
33. Modul 1.33
34. Modul 1.34
35. Modul 1.35
36. Modul 1.36
37. Modul 1.37
38. Modul 1.38
39. Modul 1.39
40. Modul 1.40
41. Modul 1.41
42. Modul 1.42
43. Modul 1.43
44. Modul 1.44
45. Modul 1.45
46. Modul 1.46
47. Modul 1.47
48. Modul 1.48
49. Modul 1.49
50. Modul 1.50
51. Modul 1.51
52. Modul 1.52
53. Modul 1.53
54. Modul 1.54
55. Modul 1.55
56. Modul 1.56
57. Modul 1.57
58. Modul 1.58
59. Modul 1.59
60. Modul 1.60
61. Modul 1.61
62. Modul 1.62
63. Modul 1.63
64. Modul 1.64
65. Modul 1.65
66. Modul 1.66
67. Modul 1.67
68. Modul 1.68
69. Modul 1.69
70. Modul 1.70
71. Modul 1.71
72. Modul 1.72
73. Modul 1.73
74. Modul 1.74
75. Modul 1.75
76. Modul 1.76
77. Modul 1.77
78. Modul 1.78
79. Modul 1.79
80. Modul 1.80
81. Modul 1.81
82. Modul 1.82
83. Modul 1.83
84. Modul 1.84
85. Modul 1.85
86. Modul 1.86
87. Modul 1.87
88. Modul 1.88
89. Modul 1.89
90. Modul 1.90
91. Modul 1.91
92. Modul 1.92
93. Modul 1.93
94. Modul 1.94
95. Modul 1.95
96. Modul 1.96
97. Modul 1.97
98. Modul 1.98
99. Modul 1.99
100. Modul 1.100

3. Handlungsschritt (25 Punkte)

Korrekturrand

- a) Im Rahmen des Projekts werden für das Kassensystem Rechner mit „multicore“-Prozessoren beschafft. Dies kann die Erstellung der Softwarekomponenten beeinflussen.

Bearbeiten Sie folgende Aufgaben mithilfe des Englischtextes.

Multi-core Programming

Multi-core processors enable multiprocessing in a single physical package. [...]

Possible gains are limited by the fraction of the software that can be run „parallised“ simultaneously on multiple cores. In the best case, so-called embarrassingly parallel problems may realize speedup factors near the number of cores.

The basic steps in designing parallel applications are:

Partitioning

This stage is intended to expose opportunities for parallel execution. The focus is on defining a large number of small tasks to get a fine grained decomposition of a problem.

Communication

The tasks generated in the partitioning phase are intended to execute concurrently but they cannot execute independently. One task will typically require data associated with another task. Data must then be transferred between tasks. This information flow is specified in the communication phase.

Agglomeration

Parallelism is so fine grained that overhead of parallel scheduling or communication swamps the useful work.

In the third stage Developers consider whether it is useful to combine, or agglomerate, tasks identified by the partitioning phase.

Mapping

In the fourth and final stage of the design of parallel algorithms, the developers specify where each task is to execute.

Vokabeln:

concurrently = gleichzeitig

to conserve = sparen

so-called embarrassingly parallel problems = hochgradig einfache parallele Probleme (alle Aufgaben sind beliebig vertauschbar)

- aa) Nennen Sie den Faktor, um den sich die Verarbeitungsgeschwindigkeit im günstigsten Fall durch parallele Datenverarbeitung mit einem Dual-core-Prozessor erhöhen lässt. 2 Punkte

- ab) Erläutern Sie die folgenden vier Schritte bei der Entwicklung von Programmen mit paralleler Datenverarbeitung. 8 Punkte

1. Partitionierung:
2. Kommunikation:
3. Zusammenfassung:
4. Zuordnung:

1. Partitionierung:

2. Kommunikation:

3. Zusammenfassung:

4. Zuordnung:

- b) Die Plan-B GmbH soll für das neue Kassensystem der Baumarkt ÖkoSpan KG die Methode *rabatt()* erstellen. Die Methode soll für einen Kundeneinkauf den gesamten gewährten Rabatt ermitteln. 15 Punkte

Korrekturrand

Vorgaben:

- Jedes Produkt ist einer Warengruppe (A, B, C ...) zugeordnet. Für jede Warengruppe ist ein Rabattsatz festgelegt. (siehe Tabelle *rabatt*). Die Tabelle soll im Laufe der Zeit um weitere Warengruppen erweitert werden.
- Die Daten der von der Kasse erfassten Waren eines Kunden werden in dem zweidimensionalen Array *artikel* vom Datentyp String gespeichert (siehe Beispiel). Auf eine Wandlung der Werte in andere Datentypen kann in der Darstellung der Logik verzichtet werden.
- Rückgabewert ist der gesamte gewährte Rabatt in EUR (*rabattGes*).

Tabelle *rabatt*

Wg_Id ¹	Rabattsatz ²
[0]	[1]
A	2,0
B	2,5
C	7,0
D	6,0
E	5,0
...	

Array *artikel* (Beispiel)

Pos	Bezeichnung	Wg_Id ¹	Listen_VKP ^{3*}	Menge
[0]	[1]	[2]	[3]	[4]
1	Dünger	A	7,80	2
2	Garteneinfassung Granit 10 x 20 x 100	B	94,86	20
3	Rosen	C	56,00	3
4	Werkzeugset	D	87,50	1
5	Fliesen, Feinsteinzeug 43 x 43	B	40,00	5
...				

¹ Wg_Id = Warengruppen-ID

² Rabattsatz (in Prozent)

³ Listen_VKP = Listenverkaufspreis in EUR

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

4. Handlungsschritt (25 Punkte)

Korrekturrand

Bisher erfasst die ÖkoSpan KG die Daten zu einem Artikel auf folgenden Formularen.

Nun wurde die Plan GmbH mit der Erstellung einer Datenbank beauftragt, in der diese Daten gespeichert werden können.

Artikelkarte		
Artikel-Nr.	Artikelbezeichnung	Lagerorte
00706	Spanplatte Ökostar (20 mm, 3 x 4 m)	2341 3421
Lieferanten		
7678		
263		

Lieferantenkarte			
Lieferer-Nr.	Firma		
7678	Spanmeier OHG		
	Straße	PLZ	Ort
	Von-Span-Allee 26	12345	A-Stadt

Artikel-Nr.	Artikelbezeichnung	Menge	Einkaufspreis
00706	Spanplatte Ökostar (20 mm, 3 x 4 m)	1 Stck.	104,00
00103	Tischlerplatte Super (18 mm, 3 x 4 m)	1 Stck.	140,00

Lagerfachkarte			
Lagerfach-Nr.	Artikel-Nr.	Mindestbestand	
2341	00706	5 Stck.	
Datum	Zugang	Abgang	Bestand
08.10.2012	20		22
16.10.2013		2	20
31.10.2013		7	13

Lagerfachkarte			
Lagerfach-Nr.	Artikel-Nr.	Mindestbestand	
3421	00706	5 Stck.	
Datum	Zugang	Abgang	Bestand
26.09.2012		10	4
14.10.2013	20		24
23.10.2013		10	14

Materialentnahmeschein			
Lagerfach-Nr.	Artikel-Nr.	Datum	Menge
2341	00706	16.10.13	2
Mitarbeiter-Nr.	Unterschrift des Mitarbeiters		
11	Scholz		

Materialentnahmeschein			
Lagerfach-Nr.	Artikel-Nr.	Datum	Menge
3421	00706	23.10.13	10
Mitarbeiter-Nr.	Unterschrift des Mitarbeiters		
45	Maschke		

- a) Erstellen Sie ein ER-Diagramm.
Geben Sie die Entitäten (keine Attribute) und deren Beziehungen mit Kardinalitäten an.

Korrekturrand
21 Punkte

- b) Der Datensatz des Lieferanten mit der Lieferanten_ID 7678 muss aus der Datenbank gelöscht werden.
Nennen Sie die Maßnahmen, mit denen referenzielle Integrität gewährleistet wird.

4 Punkte

5. Handlungsschritt (25 Punkte)

Korrekturrand

Die Plan-B GmbH nutzt zur Zeiterfassung eine Datenbank mit folgenden Tabellen:

Mitarbeiter

MA_ID	MA_Nachname	MA_Vorname	MA_GebDatum	MA_ArbeitszeitTag
811	Petermann	Friedrich	14.04.1984	8
812	Schultze	Petra	23.08.1968	4
815	Freudali	Georg	02.11.1959	8
817	Schmittmann	Udo	14.11.1992	6
841	Franzmann	Franziska	21.12.1961	8
902	Sorglos	Lisa	02.03.1954	8
...				

Zeitkonto

ZK_ID	ZK_MAID	ZK_Datum	ZK_Beginn	ZK_Ende
1	811	17.04.2013	07:00	11:45
2	811	17.04.2013	12:15	16:00
3	811	18.04.2013	07:32	08:10
4	902	17.04.2013	07:21	12:06
...				

Fehlzeit

FZ_ID	FZ_MAID	FZ_VonDatum	FZ_BisDatum	FZ_Grund	FZ_Fehltage
1	811	18.04.2013	23.04.2013	Nicht anwesend	4
2	902	18.04.2013	08.05.2013	Krank	14
3	811	19.06.2013	20.06.2013	Krank	2
4	811	17.11.2013	17.11.2013	Nicht anwesend	1
5	904	31.12.2013	31.12.2013	Nicht anwesend	1
6	904	01.01.2014	09.01.2014	Nicht anwesend	6

Hinweis: Jahresübergreifender „Nicht anwesend“ generiert zwei Datensätze (siehe Beispiel Tabelle *Fehlzeit* FZ_MAID 904).

a) Die bestehende Datenbank soll wie im Folgenden beschrieben verändert werden.

Erstellen Sie dazu jeweils die SQL-Anweisung.

aa) Die Tabelle *Fehlzeit* soll gelöscht werden.

3 Punkte

ab) Die Datenbank wurde um die Nachschlagetabelle *Fehlzeitgrund* ergänzt.

Fehlzeitgrund

FZG_ID	FZG_Bezeichnung
1	Nicht anwesend
2	Krank

Die neue Tabelle *Fehlzeit* soll dementsprechend folgende Struktur erhalten.

Fehlzeit

FZ_ID	FZ_MAID	FZ_VonDatum	FZ_BisDatum	FZ_FZGID	FZ_Fehltage
1	811	18.04.2013	23.04.2013	1	4
2	902	18.04.2013	08.05.2013	2	14
...					

In die Spalte *FZ_FZGID* sollen nur Werte eingetragen werden können, die in der Tabelle *Fehlzeitgrund* als Primärschlüssel vorkommen.

In der Spalte *FZ_MAID* sollen nur solche Werte eingetragen werden können, die in der Tabelle *Mitarbeiter* als Primärschlüssel vorkommen

Erstellen Sie eine SQL-Anweisung, mit der die neue Tabelle *Fehlzeit* entsprechend der Vorgaben erstellt werden kann. 6 Punkte

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

SQL-Syntax-Referenz (Auszug)

Syntax	Beschreibung
Tabellen	
CREATE TABLE Tabellennamen (Feldname < DATENTYP >, Primärschlüssel, Fremdschlüssel)	Erzeugt eine neue leere Tabelle mit der beschriebenen Struktur
CHARACTER	Textdatentyp
DECIMAL	Numerischer Datentyp (Festkommazahl)
DOUBLE	Numerischer Datentyp (Doppelte Präzision)
INTEGER	Numerischer Datentyp (Ganzzahl)
DATE	Datum (Format DD.MM.YYYY)
PRIMARY KEY	
FOREIGN KEY (Feldname) REFERENCES	
DROP TABLE Tabellennamen	Löscht eine Tabelle
Befehle, Klauseln, Attribute	
SELECT * Feldname1 [, Feldname2, ...]	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
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 Beispiel: FROM Verkaufer LEFT JOIN Kunde ON Verkaufer.Ver_ID = Kunde.Ver_ID
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 Beispiel: FROM Verkaufer RIGHT JOIN Kunde ON Verkaufer.Ver_ID = Kunde.Ver_ID
FULL JOIN	Liefert aus beiden Tabellen jeweils alle Datensätze
WHERE	Bedingung, nach der Datensätze ausgewählt werden sollen Beispiel: WHERE name = 'Maier'
GROUP BY Feldname1 [,Feldname2, ...]	Gruppierung (Aggregation) nach Inhalt des genannten Feldes Beispiel: GROUP BY name, vorname
ORDER BY Feldname1 [,Feldname2, ...] ASC DESC	Sortierung nach Inhalt des genannten Feldes oder der genannten Felder ASC: aufsteigend; DESC: absteigend Beispiel: ORDER BY name ASC
Datenmanipulation	
DELETE FROM Tabellennamen	Löschen von Datensätzen in der genannten Tabelle
UPDATE Tabellennamen SET	Aktualisiert Daten in Feldern einer Tabelle Beispiel: UPDATE Artikel SET (Preis=10.00)
INSERT INTO Tabellennamen ... 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 Beispiele: INSERT INTO kunde VALUES 56532, 'Martina', 'Schmitz', '12345', 'Berlin' INSERT INTO kunde SELECT * FROM vertrag WHERE stadt='Berlin'
Funktionen	
AVG (Feldname)	Ermittelt das arithmetische Mittel aller Werte im angegebenen Feld
COUNT (Feldname *)	Ermittelt die Anzahl der Datensätze mit Nicht-NULL-Werten im angegebenen Feld oder alle Datensätze der Tabelle (dann mit Operator *)
SUM (Feldname Formel)	Ermittelt die Summe aller Werte im angegebenen Feld oder der Formelergebnisse Beispiel: SELECT SUM(preis)
MIN (Feldname Formel)	Ermittelt den kleinsten aller Werte im angegebenen Feld Beispiel: SELECT MIN(preis)
MAX (Feldname Formel)	Ermittelt den größten aller Werte im angegebenen Feld Beispiel: SELECT MAX(preis)
Datumsfunktionen	
CURRENT	Liefert das aktuelle Datum mit der aktuellen Uhrzeit
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
Operatoren	
AND	Logisches UND
NOT	Logische Negation
OR	Logisches ODER
=	Test auf Gleichheit
>, >=, <, <=, < >	Test auf Ungleichheit
*	Multiplikation
/	Division
+	Addition, positives Vorzeichen
-	Subtraktion, negatives Vorzeichen

- b) Erstellen Sie eine SQL-Abfrage, welche die Urlaubstage aller Mitarbeiter im Jahr 2013 ermittelt und nach ermittelten Urlaubstagen sortiert.

10 Punkte

Beispielausgabe

811	Petermann	Friedrich	0
812	Schultze	Petra	0
815	Freudali	Georg	13
817	Schmittmann	Udo	18
841	Franzmann	Franziska	21
...			

Fortsetzung 5. Handlungsschritt

Korrekturrand

- c) Für den Mitarbeiter Friedrich Petermann wurde in der Tabelle *Fehlzeit* ein Datensatz wie folgt falsch erfasst:
Statt einer zweitägigen „Dienstreise“ für den 17.11. und 18.11.2013 wurde versehentlich nur für den 17.11.2013 ein eintägiger „Urlaub“ eingetragen (siehe Tabelle *Fehlzeit*).

Erstellen Sie eine SQL-Anweisung, mit der die Korrektur durchgeführt werden kann.

6 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.

☐