Korrekturrand

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

Sie sind Mitarbeiter/-in der Systemhaus GmbH.

Die Systemhaus GmbH wurde von der Sportsfit AG mit der Programmierung einer Vereinsverwaltungs-Software beauftragt.

Sie arbeiten in diesem Projekt mit.

Sie sollen im Rahmen des Projekts vier der folgenden fünf Aufgaben erledigen:
1. Ein Teilprojekt planen
2. Eine Methode programmieren (prozedurale Programmierung)
3. Für die objektorientierte Entwicklung einer Geräte-Steuerung UML-Diagramme erstellen
4. Ein Datenbankmodell in der dritten Normalform entwickeln
5. Für eine bestehende Datenbank mehrere SQL-Anweisungen erstellen

1. Handlungsschritt (25 Punkte)

Sie sind an der Planung eines Teilprojekts beteiligt.

a) Für das Teilprojekt liegen ein Lastenheft und ein Pflichtenheft vor.

Beschreiben Sie in folgender Tabelle Lastenheft und Pflichtenheft. 6 Punkte

	Lastenheft	**Pflichtenheft**
Verfasser		
Verwendung		
Inhalt		

b) Das Projekt soll mithilfe eines GANTT-Diagramms und eines Netzplans geplant werden.

ba) Nennen Sie drei Informationen, die Sie in einem GANTT-Diagramm darstellen können. 3 Punkte

bb) Nennen Sie drei Informationen, die Sie nur einem Netzplan entnehmen können. 3 Punkte

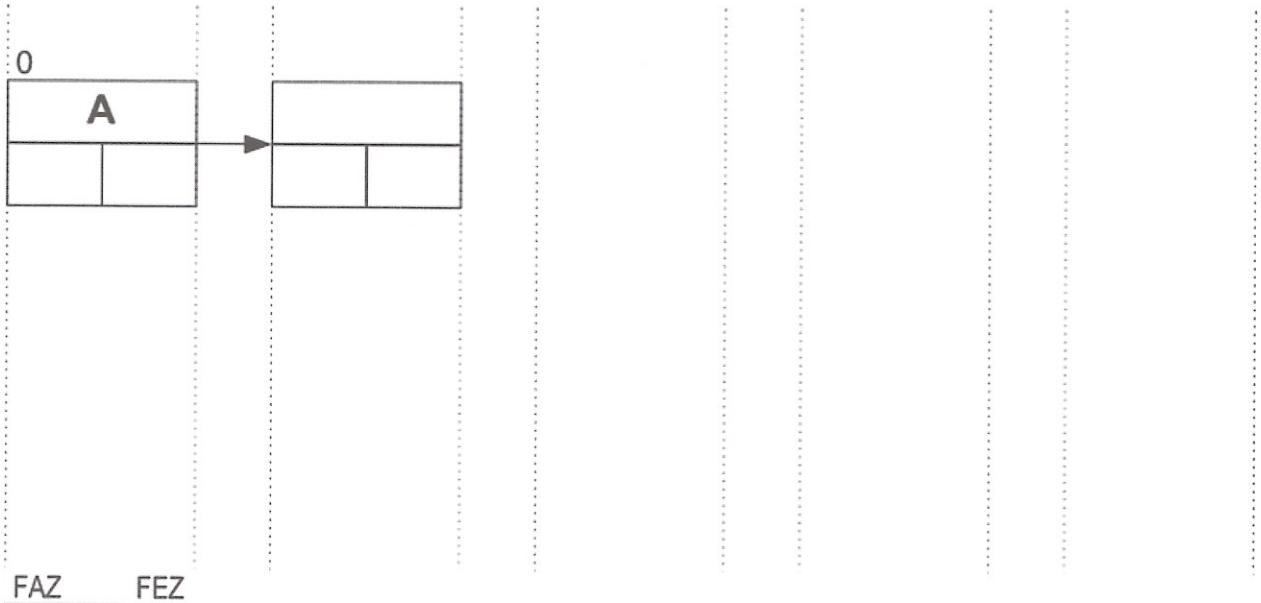
c) Für ein Teilprojekt wurden folgende Vorgänge geplant, die von den genannten Mitarbeitern erledigt werden sollen.

ca) 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

Vorgang	Beschreibung	Dauer	Vorgänger
A	Planung	3	-
B	Softwareentwicklung	7	A
C	Datenbankentwicklung	4	A
D	Testphase	1	B, C
E	Installation, Integration	2	D
F	Übergabe, Abnahme	1	E

Netzplan für das Teilprojekt „< Name >"

Korrekturrand



FAZ: Frühester Anfangszeitpunkt
FEZ: Frühester Endzeitpunkt
SAZ: Spätester Anfangszeitpunkt
SEZ: Spätester Endzeitpunkt
GP: Gesamtpuffer

cb) Die Übergabe des Teilprojekts soll spätestens am Freitag, 16.12.2016, erfolgen. Samstags und sonntags wird nicht gearbeitet.

Ergänzen Sie anhand des folgenden Personaleinsatzplans den nachstehenden Projektplan für dieses Teilprojekt im vorbereiteten Gantt-Diagramm (siehe Beispiel Vorgang A).
Markieren Sie die Tage der Vorgänge jeweils mit X. 5 Punkte

Personaleinsatzplan

	November 2016																	Dezember 2016															
	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	14	15	16	17	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
Dr. Huber			A	A	A																											F	
Fischer			A	A	A																	D				E			E			F	
Kramer										C	C	C			C																		
Müller								B	B	B	B	B			B	B										E			E				
Schneider			A	A	A			B	B	B	B	B			B	B						D										F	

Projektplan (Gantt-Diagramm)

	November 2016																	Dezember 2016															
	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	14	15	16	17	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
A Planung			X	X	X																												
B SW-Entw.																																	
C DB-Entw.																																	
D Test																																	
E Inst./Int.																																	
F Übergabe																																	

2. Handlungsschritt (25 Punkte)

Ko

Die Systemhaus GmbH soll ein Programm entwickeln, mit dem die Sportsfit AG Rechnungen für Vereinsmitglieder erstellen kann. Die abzurechnenden Leistungen sind in einem Journal gespeichert.

Journal (Beispiel)

Datum	KundenID	LeistungsID	EinzelPreis	Anzahl
01.04.2016	K00091	100076	2,40	2
10.04.2016	K00091	100076	2,40	3
10.04.2016	K00091	500123	15,00	1
03.04.2016	K01234	200234	20,00	1
11.04.2016	K01234	200234	20,00	1
05.04.2016	K01234	200356	15,00	1

Das Journal ist nach KundenID und bei gleicher KundenID nach LeistungsID sortiert.

Die Rechnungen sollen wie folgt aufgebaut sein:

K00091

Pos	LeistungsID	Bezeichnung der Leistung	Anzahl	Einzelpreis	Gesamtpreis
1	100076	Softgetränke	5	2,40	12,00
2	500123	Buch, „Fit ohne Anstrengung"	1	15,00	15,00
				Rechnungssumme (netto)	27,00

K01234

Pos	LeistungsID	Bezeichnung der Leistung	Anzahl	Einzelpreis	Gesamtpreis
1	200234	Kraftraumstunde	2	20,00	40,00
2	200356	Saunastunde	1	15,00	15,00
				Rechnungsbetrag (netto)	55,00

Hinweis:
Weitere auf Rechnungen übliche Angaben, wie z. B. die Adresse, Mehrwertsteuer, sollen in diesem Entwurf nicht berücksichtigt werden.

Folgende Funktionen sollen verwendet werden:

hole_journalsatz() : String	Liest den nächsten Datensatz der Journal-Tabelle in eine Zeichenkette ein. Kann kein Satz mehr gelesen werden, liefert die Funktion den String "".
lese_kundenid(satz : String) : Integer	Ermittelt die *KundenID* innerhalb *satz*
lese_leistungsid(satz : String) : Integer	Ermittelt die *LeistungsID* innerhalb *satz*
lese_einzelpreis(satz : String) : Double	Liefert den *Einzelpreis* innerhalb *satz*
lese_anzahl(satz : String) : Integer	Ermittelt die Anzahl (der Leistungen) innerhalb *satz*
hole_bezeichnung(leistungsid : Integer) : String	Liefert die Bezeichnung der Leistung
schreibe_kundenid(kundenid : Integer)	Schreibt die *KundenID*
schreibe_kopfzeile()	Schreibt die Kopfzeile der Positionen-Tabelle: Pos LeistungsID Bezeichnung der Leistung Anzahl Einzelpreis Gesamtpreis
schreibe_positionszeile(pos : Integer, leistungsid : Integer, bezeichnung : String, anzahl : Integer, einzelpreis : Double gesamtpreis : Double)	Schreibt die Zeile für eine Position in der geforderten Darstellung. Der Gesamtpreis einer Position wird von der Prozedur berechnet.
schreibe_rechnungssumme(rechnungssumme : Double)	Schreibt die Rechnungssumme

Entwickeln Sie auf der Folgeseite einen Algorithmus für die Prozedur *erstelleRechung()*.

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

PR
Wie
1
ZPA

3. Handlungsschritt (25 Punkte)

Korrektu

In der Sportsfit AG werden verschiedene Geräte zum Herz-Kreislauf-Training eingesetzt.

Die Systemhaus GmbH soll für diese Geräte eine Steuerung entwickeln, die folgendes leistet:

- 3 Leistungsstufen
- Start mit Leistungsstufe 1
- Erhöhung der Leistungsstufe nach einer Minute, wenn Pulsfrequenz < maximale Pulsfrequenz
- Reduzierung der Leistungsstufe, wenn Pulsfrequenz > maximale Pulsfrequenz
- Abschalten, wenn Trainingszeit abgelaufen ist
- Pulsfrequenz messen (Methode pulsMessen())

a) Vervollständigen Sie das folgende unvollständige UML-Zustandsdiagramm. 10 Punkte

Hinweis: Notation für UML-Zustandsdiagramm siehe Belegsatz, Seite 2

b) Erstellen Sie anhand der Vorgaben das UML-Klassendiagramm für die Klasse *Geraet*. Korrekturrand

Nennen Sie dazu alle erforderlichen Attribute und Methoden mit den entsprechenden Datentypen. 7 Punkte

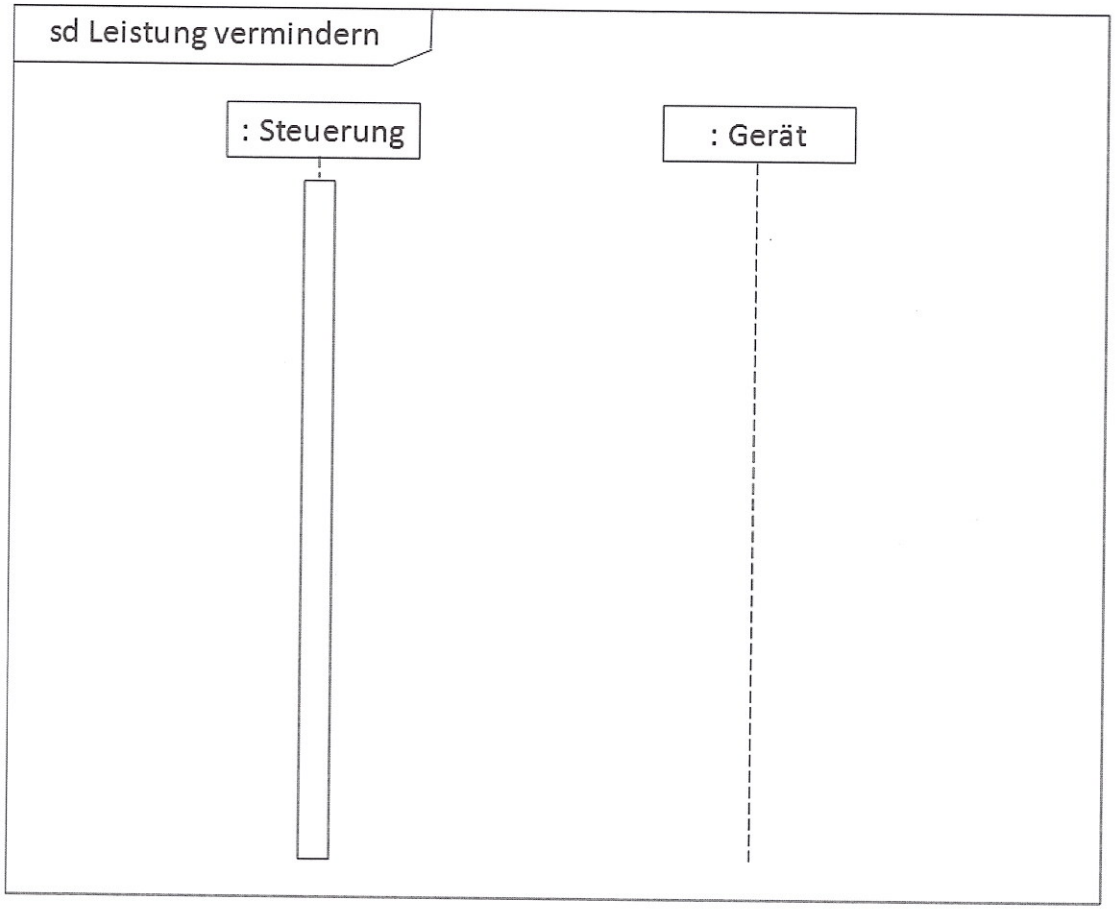
Hinweis: Notation für UML-Klassendiagramm siehe Belegsatz, Seite 3

c) Die Steuerung fragt alle fünf Sekunden die Pulsfrequenz vom Gerät ab.

- Wenn die gemessene Pulsfrequenz größer als die festgelegte maximale Pulsfrequenz ist, löst die Steuerung eine Verminderung der Leistungsstufe im Gerät aus.
- Sonst bleibt die momentane Leistungsstufe erhalten.

Modellieren Sie dieses Szenario in einem UML-Sequenzdiagramm, dessen Grundstruktur wie folgt gegeben ist. 8 Punkte

Hinweis: Notation für UML-Sequenzdiagramm siehe Belegsatz, Seite 4



4. Handlungsschritt (25 Punkte)

Korrektur

Bislang hat die Sportsfit AG ihre Getränkebestellungen in folgender Tabelle verwaltet.

Bestell-Nr.	Datum	Lieferant	Artikel	Menge	Einzelpreis
1	10.02.2016	Schiller KG, Goetheplatz 3, 12345 Berlin	Hydro-Wasser Cola-Mix-Spree	5 Kisten 7 Kisten	7,99 EUR 11,99 EUR
2	17.02.2016	Maurer GmbH, Bahnweg 45, 2345 Potsdam	Stone Energy-Drink	10 Kartons	22,00 EUR
3	24.02.2016	Baecker GmbH, Platz 45, 65623 Reuth	Hydro-Wasser Stone-Energy-Drink	10 Kisten 4 Kartons	5,46 EUR 20,00 EUR
4	10.03.2016	Schiller KG, Goetheplatz, 12345 Berlin	„Weizen Hell Rhein-Bräu"	12 Kisten	13,99 EUR
5	16.03.2016	Bäcker GmbH, Platz 45, 65623 Reuth	Hydro-Wasser Stone-Energy-Drink	10 Kisten 4 Kartons	5,46 EUR 20,00 EUR
6	23.03.2016	Bäkker GmbH, Platz 45, 65623 Reut	Hydro-Wasser Stone-Energy-Drink	10 Kisten 4 Kartons	5,46 EUR 20,00 EUR
...					

Die Systemhaus GmbH soll nun ein Bestellsystem entwickeln. In diesem Zusammenhang muss ein relationales Datenbankmodell entwickelt werden.

a) Erläutern Sie, was eine erfolgreiche Normalisierung in die dritte Normalform in einem relationalen Datenbankmodell durch Vermeidung von Redundanzen verhindert.

3 Punkte

b) Entwickeln Sie auf der Folgeseite ein entsprechendes relationales Datenbankmodell in der dritten Normalform.

- Entwerfen Sie die dazu erforderlichen Tabellen.
- Kennzeichnen Sie Primärschlüssel mit PK und Fremdschlüssel mit FK.
- Geben Sie die Beziehungen mit den Kardinalitäten an.

Hinweis: PLZ und Ort dürfen in einer Tabelle vorhanden sein.

22 Punkte

5. Handlungsschritt (25 Punkte)

Korrek

Die Systemhaus GmbH hat für die Sportsfit AG folgende Datenbank zur Verwaltung von Kursen entwickelt.

Kunde
KundenID (PK)
Nachname
Vorname
Straße
Plz
Ort

KundeKurs
KundenID (FK)
KursID (FK)

Sportart
- SportartID (PK)
- Bezeichnung

Kursart
KursartID (PK)
Bezeichnung
SportartID (FK)
TeilnehmerMax

Kurs
KursID (PK)
KursartID (FK)
TrainerID (FK)
SportstaetteID (FK)
DatumUhrzeitBeginn
Preis

Trainer
TrainerID (PK)
Name

TrainerSportart
TrainerID (FK)
SportartID (FK)

Sportstaette
SportstaetteID (PK)
Bezeichnung

Für diese Datenbank sollen Sie einige SQL-Anweisungen erstellen.

Hinweis: Die *TeilnehmerMax* in der Tabelle *Kursart* ist immer größer 0.
SQL-Syntax (Auszug) siehe Belegsatz, Seite 4-5

a) Gewünschte Datenänderung:
- Reduzierung der Preise um 10 % aller Kurse, die vor 16:00 Uhr beginnen

Erstellen Sie die entsprechende SQL-Anweisung. 4 Punkte

b) Gewünschte Ausgabe:
- Liste aller Kunden aus dem Postleitzahlbereich ‚50000' bis ‚59999'
- Auflistung aller Kundenattribute
- Aufsteigende Sortierung nach Nachname, Vorname

Erstellen Sie die entsprechende SQL-Anweisung. 5 Punkte

Hinweis:
Die Postleitzahlen liegen in der Datenbank alphanumerisch vor.

Korrekturrand

c) Gewünschte Ausgabe:
– Liste der Kurse, die weniger als 50 Prozent ausgelastet sind

Siehe Beispiel:

KursID	Bezeichnung	DatumUhrzeitBeginn	Auslastung
K767	Joggen	04.05.2016 17:00	43,5
K676	Volleyball	13.09.2016 19:00	28,6
...			

Erstellen Sie die entsprechende SQL-Anweisung.

9 Punkte

d) Gewünschte Ausgabe:
– Liste der angebotenen Sportarten mit Anzahl der dafür verfügbaren Trainer

Siehe Beispiel:

SportartID	Bezeichnung	AnzahlTrainer
3	Joggen	5
5	Fussball	2
...		

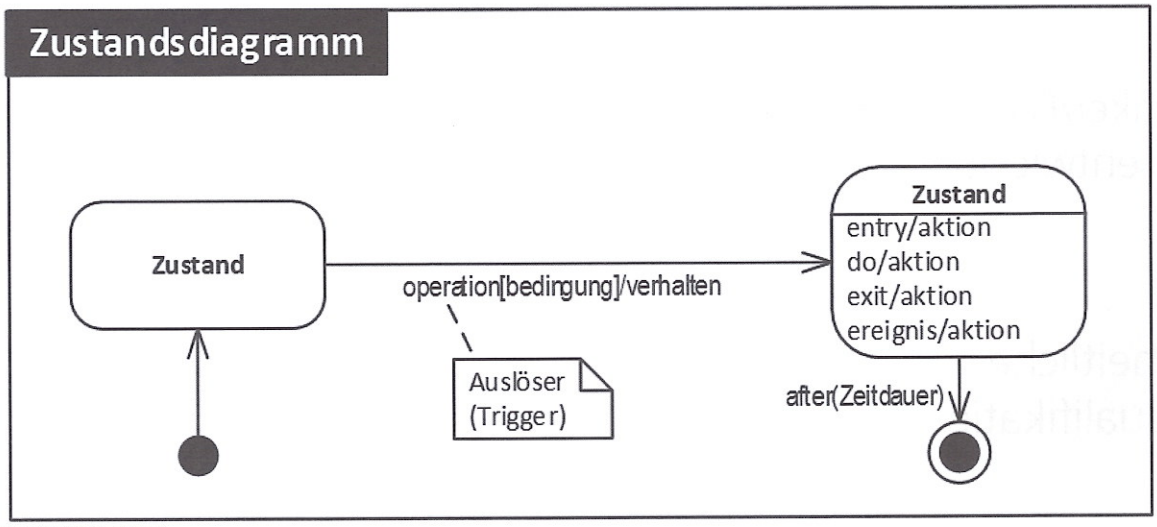
Erstellen Sie die entsprechende SQL-Anweisung.

7 Punkte

bitte wenden!

UML-Zustandsdiagramm, Notation (Auszug)

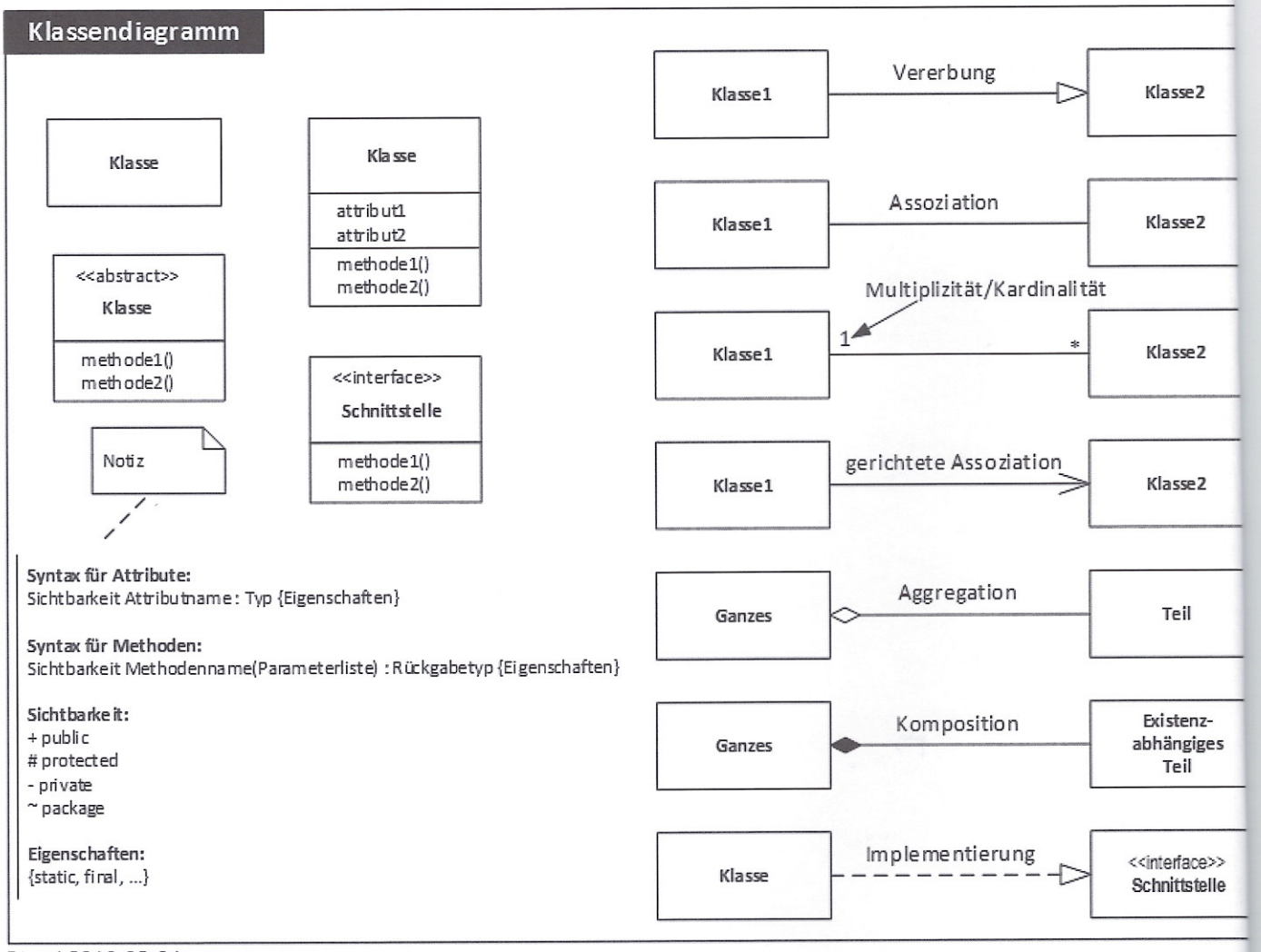
für 3. Handlungsschritt, Teilaufgabe a)



Stand 2016-02-04

UML-Klassendiagramm, Notation

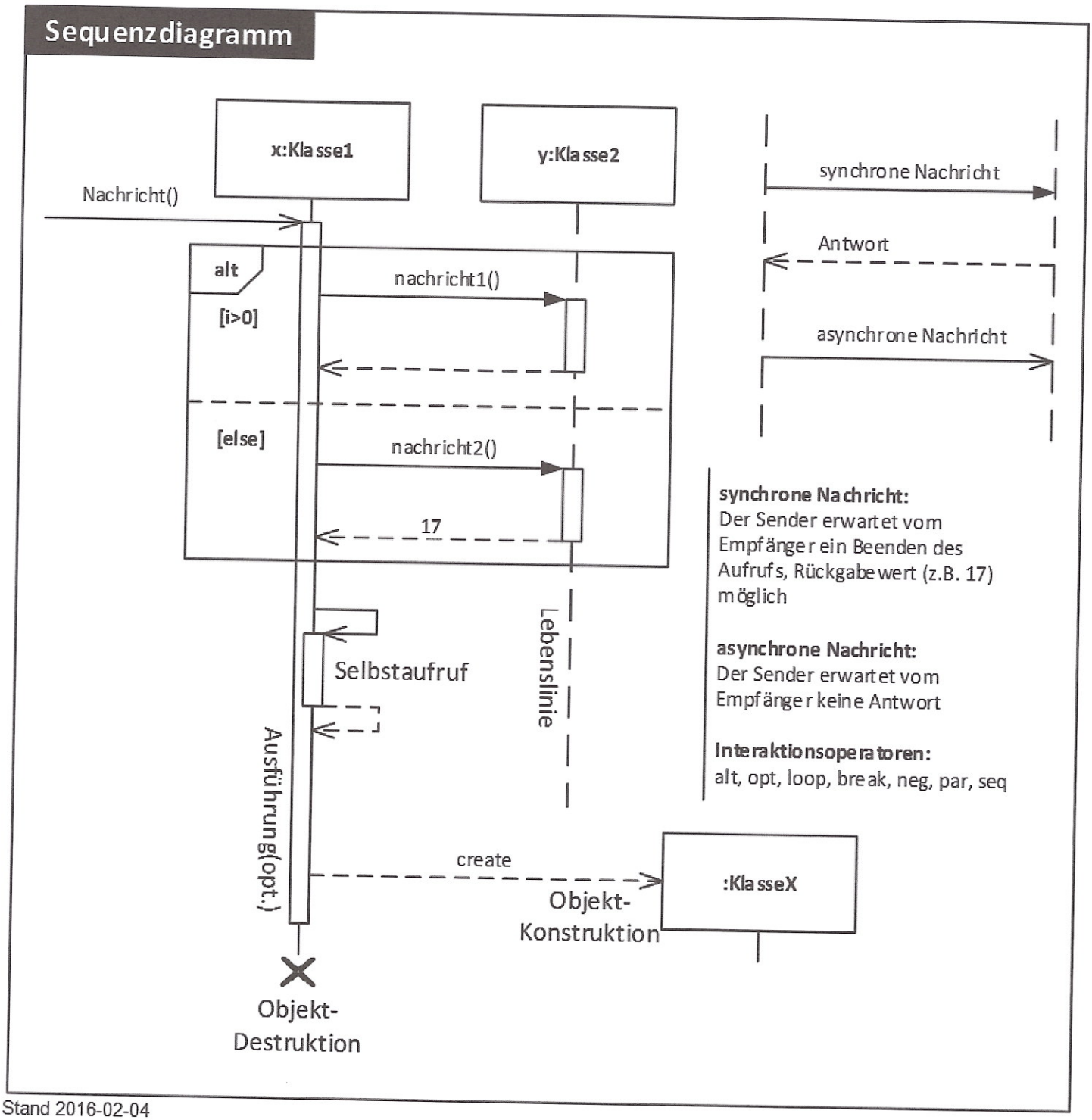
für 3. Handlungsschritt, Teilaufgabe b)



Stand 2016-02-04

UML-Sequenzdiagramm, Notation (Auszug)

für 3. Handlungsschritt, Teilaufgabe d)



Stand 2016-02-04

SQL-Syntax (Auszug)

für 5. Handlungsschritt

Syntax	Beschreibung
Tabelle	
CREATE TABLE Tabellenname(Feldname < DATENTYP >, Primärschlüssel, Fremdschlüssel)	Erzeugt eine neue leere Tabelle mit der beschriebenen Struktur
ALTER TABLE Tabellenname **ADD COLUMN** Spaltenname Datentyp **DROP COLUMN** Spaltenname	Änderungen in eine Tabelle: Hinzufügen einer Spalte Entfernen einer Spalte
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 Tabellenname	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
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
GROUP BY Feldname1 [,Feldname2, ...]	Gruppierung (Aggregation) nach Inhalt des genannten Feldes
ORDER BY Feldname1 [,Feldname2, ...] **ASC | DESC**	Sortierung nach Inhalt des genannten Feldes oder der genannten Felder ASC: aufsteigend; DESC: absteigend
Datenmanipulation	
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
Aggregatfunktionen	
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
MIN(Feldname | Formel)	Ermittelt den kleinsten aller Werte im angegebenen Feld
MAX (Feldname | Formel)	Ermittelt den größten aller Werte im angegebenen Feld

SQL-Syntax (Auszug) – Fortsetzung

Syntax	Beschreibung
Funktionen	
LEFT(Zeichenkette, Anzahlzeichen)	Liefert *Anzahlzeichen* der Zeichenkette von links.
RIGHT(Zeichenkette, Anzahlzeichen)	Liefert *Anzahlzeichen* 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)
Operatoren	
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 2016-02-04