

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

Die Firma Speiche GmbH betreibt einen Fahrradverleih mit Werkstatt.

Sie arbeiten in der EProg GmbH, die Softwarelösungen für Handel und Dienstleistungen zur Verfügung stellt und verwaltet.

Sie sollen vier der folgenden fünf Aufgaben in diesem Projekt erledigen:

1. Beim Management für das Projekt Abrechnungssoftware mitwirken
2. Programm zur Auswertung der Arbeitszeiterfassung anfertigen
3. Objektorientierte Software für Ladegerät entwickeln
4. Tabelle Wartung normalisieren
5. SQL-Abfragen zur Verleihdatenbank formulieren

1. Handlungsschritt (25 Punkte)

Für die Abrechnung der Servicemitarbeiter der Speiche GmbH soll eine Abrechnungssoftware eingeführt werden.

a) Sie erhalten den Auftrag, eine Anforderungsanalyse für diese Software durchzuführen.

aa) Nennen Sie zwei Methoden, die Sie für eine Anforderungsanalyse anwenden können.

2 Punkte

ab) Beschreiben Sie zwei Anforderungen an die neu einzuführende Software.

4 Punkte

b) Der Projektleiter Ihres Teams hat Ihnen mitgeteilt, dass das Projekt „Abrechnungssoftware“ mit einer Kick-off-Sitzung begonnen wird.

ba) Nennen Sie jeweils vier auf der Sachebene und der Beziehungsebene liegende Aufgabenstellungen dieser Kick-off-Sitzung.

8 Punkte

Sachebene	Beziehungsebene

bb) Nennen Sie drei Befugnisse, die der Projektleiter zur Wahrnehmung seiner Leitungsaufgaben haben muss.

3 Punkte

Korrekturrand

c) Ihr Projektteam ist mit der Auswahl weiterer Softwarekomponenten beauftragt. Nach einer umfangreichen Marktsondierung haben Sie Ihre Auswahl auf zwei Softwarelösungen begrenzt. Um eine endgültige Entscheidung zu treffen, sollen Sie die Alternativen in einer Nutzwertanalyse vergleichen.

ca) Ergänzen Sie die nachfolgende Tabelle zur Nutzwertanalyse um weitere fünf Kriterien mit sinnvoller Gewichtung. Vervollständigen Sie die Tabelle mit sinnvollen Beispielwerten für die Anbieter A und B. Entscheiden Sie anhand Ihrer Daten, welcher Anbieter den Zuschlag erhalten soll.

6 Punkte

Kriterium	Gewichtung (G)	Anbieter A		Anbieter B	
		Erfüllung (E)	Nutzwert (N)	Erfüllung (E)	Nutzwert (N)
Image des Softwareanbieters	25	1	25	3	75
SUMME					

cb) Nennen Sie einen möglichen Kritikpunkt an der Nutzwertanalyse.

2 Punkte

2. Handlungsschritt (25 Punkte)

Korrekturrand

Die Verleihfirma 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 Gehen-Zeit. (Pausen werden nicht berücksichtigt.)

Die Zeiterfassungsliste, die alle Buchungen eines Mitarbeiters für einen Monat anzeigt, soll wie folgt aufgebaut werden (siehe auch Beispiel).

- Liegen für einen Tag die Kommen- und Gehen-Buchungen 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 Kommen-Zeit, 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 Gehen-Zeiten eines Mitarbeiters für einen Monat liegen in einer zweidimensionalen Zeiterfassungstabelle vor.

Beispiel

Zeiterfassungsliste

Mitarbeiter: 12345		Oktober 2019		
Tag	Kommen	Gehen	Anwesenheit	Bemerkung
1			00:00	nicht anwesend
2	08:10	17:20	09:10	
3	07:50		00:00	Buchung fehlt
4			00:00	nicht anwesend
5			00:00	nicht anwesend
6	08:00	16:00	08:00	
7	16:30		00:00	Buchung fehlt
8	08:20	16:40	08:20	
30	08:10		00:00	Buchung fehlt
31			00:00	nicht anwesend
Summe Anwesenheit:				43:10

Zeiterfassungstabelle

Tag	Stunde	Minute
2	8	10
2	17	20
3	7	50
6	8	00
6	16	00
7	16	30
8	8	20
8	16	40
30	8	10

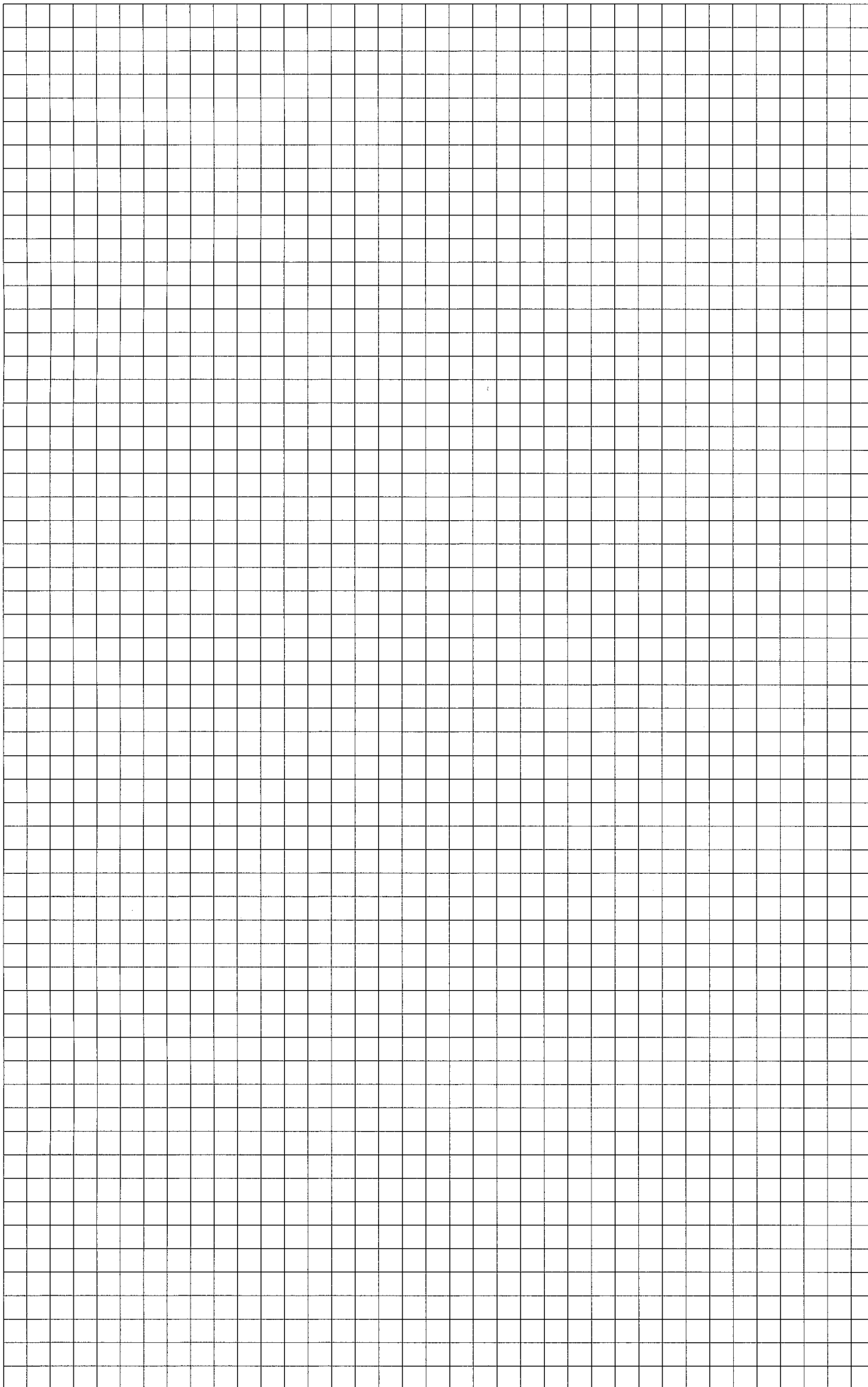
Erstellen Sie für die Methode ,erzeugeListe()' einen entsprechenden Algorithmus in Pseudocode, Struktogramm oder PAP.

Folgende Funktionen sind bereits implementiert:

tagEImMonat(monat : int, jahr : int) : int	Ermittelt die Anzahl der Tage für den übergebenen Monat eines Jahres.
schreibeKopf(persnr : int, jahr : int, monat : int)	Gibt die Kopfzeilen der Liste aus.
schreibeZeile(tag : int, std1 : int, min1 : int, std2 : int, min2 : int, anwTag : int, bemerkung : String)	Gibt eine Datenzeile aus. Für fehlende Zeiten ist der Wert -1 anzugeben. Die Tagesanwesenheit wird der Funktion in Minuten übergeben und von ihr in Stunden:Minuten ausgegeben.
schreibeFuss(anwMonat : int)	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



3. Handlungsschritt (25 Punkte)

Korrekturrand

Bei der Speiche GmbH werden programmierbare Ladegeräte für E-Bike-Akkus eingesetzt.

Sie sollen als Mitarbeiter/-in der EProg GmbH eine Software entwickeln, die folgendes leistet:

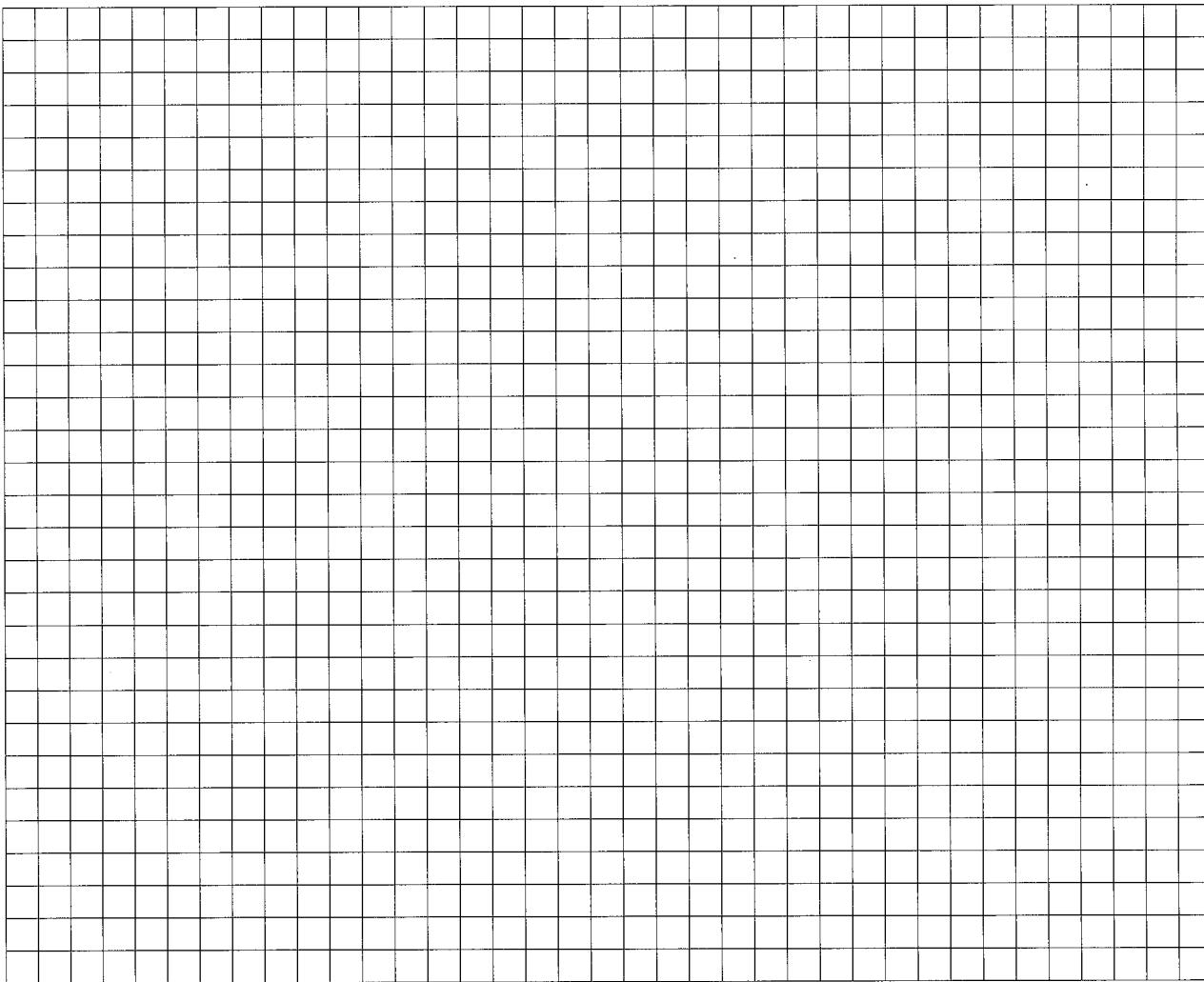
- Nach dem *Einschalten* befindet sich das Ladegerät im Zustand *nichtLadend*.
- Ist der Ladezustand des Akkus *kleiner 20 Prozent*, dann ist der Akku defekt. Das Ladegerät bleibt im Zustand *nichtLadend*.
- Ist der Ladezustand des Akkus *größer gleich 20 und kleiner 100 Prozent*, dann schaltet das Gerät zunächst in den Zustand *normalLadend*.
- Ist der Ladezustand *kleiner 80 Prozent* wird in den Zustand *schnellLadend* weitergeschaltet.
- Sobald der Ladezustand *80 Prozent erreicht*, schaltet das Gerät in den Zustand *normalLadend* zurück.
- Ist der Ladezustand von *100 Prozent erreicht*, dann wechselt das Gerät wieder in den Zustand *nichtLadend* und verbleibt in diesem.

Hinweis: Die Auslösung für Zustandsübergänge (Transitionen) erfolgt minütlich.

a) Erstellen Sie zum oben beschriebenen Ladevorgang ein entsprechendes UML-Zustandsdiagramm.

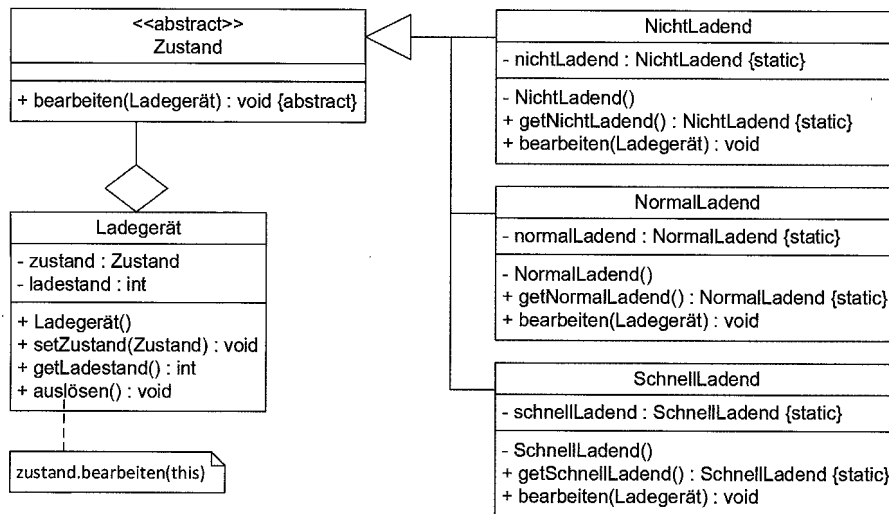
16 Punkte

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



- b) Die Software soll objektorientiert nach dem Zustand-Entwurfsmuster (State Pattern) programmiert werden. Für die Umsetzung liegt das folgende UML-Klassendiagramm vor.

Korrekturrand



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

- ba) Im Konstruktor der Klasse *Ladegerät* wird der Anfangszustand durch Initialisierung von *zustand* mit einem *NichtLadend*-Objekt festgelegt.

Formulieren Sie die entsprechende Anweisung.

2 Punkte

+ Ladegerät()

- bb) In der *bearbeiten*-Methode der Klasse *NichtLadend* wird bei einem Akku-*ladestand* größer gleich 20 und kleiner 100 der Referenz *zustand* des Ladegeräts ein *NormalLadend*-Objekt zugewiesen.

Formulieren Sie die Kontrollstruktur mit entsprechender Anweisung.

3 Punkte

+ bearbeiten(ladegerät : Ladegerät) : void

- bc) Erläutern Sie anhand des gegebenen Entwurfsmusters den Begriff Polymorphie. Nutzen Sie dazu die Instanzvariable *zustand*.

4 Punkte

4. Handlungsschritt (25 Punkte)

Korrekturrand

Der nachfolgende Tabellenausschnitt zeigt, wie in der Werkstatt der Speiche GmbH die Wartung(*Wart*) der Räder(*Rad*) durch Mitarbeiter(*Ma*) dokumentiert wird.

Sie sollen als Mitarbeiter/-in der EProg GmbH diesen Datenbestand in drei Schritten in eine relationale Datenbank überführen.

Dazu liegen folgende Informationen vor:

RadID	RadTyp	WartDatum	WartArtID	WartArt	Zeit	MaID	MaName
E5	E-Bike 400	2019-10-17	12, 09, 05	Bremse Schaltung, Akku	30 12 15	123, 345, 456	Klaus Müller, Beatrice Richter, Kurt Helmig
C2	Citybike 28	2019-10-20	03, 12	Lager, Bremse	25 10	345, 123	Beatrice Richter Klaus Müller
E5	E-Bike 400	2019-11-15	09	Schaltung	15	123	Klaus Müller
First Normal Form		Table contains only atomic values					
Second Normal Form		No non-key attributes depend on a portion of the primary key (all fields in a table are entirely related to a primary key)					
Third Normal Form		Table contains only columns that are non-transitively dependent on the primary key					

- a) Erstellen Sie auf der gegenüberliegenden Seite aus der gegebenen Tabelle eine neue Tabelle, die der ersten Normalform entspricht. 9 Punkte

Tragen Sie alle Attributwerte ein. Bilden Sie aus den bestehenden Attributen einen zusammengesetzten Primärschlüssel. Kennzeichnen Sie die einzelnen Teilattribute des Primärschlüssels durch unterstreichen.

- b) Bringen Sie den Datenbestand durch Aufteilung in mehrere Tabellen in die zweite Normalform. Geben Sie den Tabellen sinnvolle Namen. Kennzeichnen Sie die Primärschlüssel in den Tabellen durch unterstreichen. Geben Sie die Beziehungen zwischen den Tabellen an. 11 Punkte

This image shows a full page of blank graph paper. The grid consists of small, equal-sized squares formed by thin black lines. There are no margins, text, or other markings on the page.

zu a)

RadID	RadTyp	WartDatum	WartArtID	WartArt	Zeit	MaID	MaName
E5	E-Bike 400	2019-10-17	12, 09, 05	Bremse Schaltung, Akku	30 12 15	123, 345, 456	Klaus Müller, Beatrice Richter, Kurt Helmig
C2	Citybike 28	2019-10-20	03, 12	Lager, Bremse	25 10	345, 123	Beatrice Richter Klaus Müller
E5	E-Bike 400	2019-11-15	09	Schaltung	15	123	Klaus Müller

Hinweis: Die Anzahl der Zeilen und Spalten der Leertabelle geben keinen Hinweis auf die Lösung.

Hinweis: Ab hier können die Attributwerte weggelassen werden.

Korrekturrand

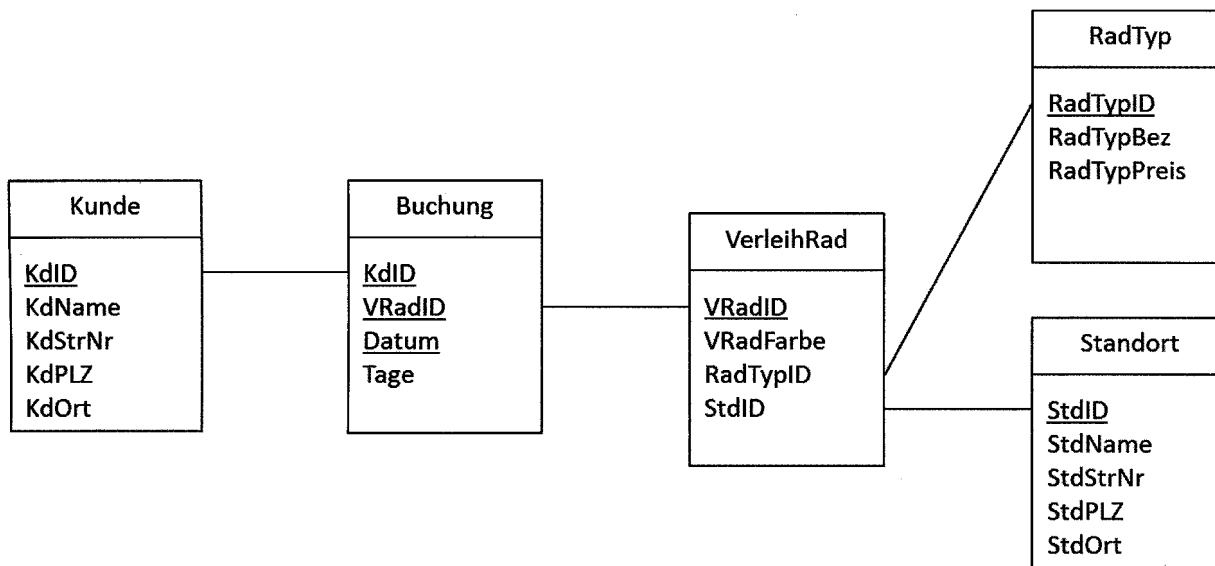
- 5 Punkte

[illegible]

5. Handlungsschritt (25 Punkte)

Korrekturrand

Die Speiche GmbH verwaltet ihre Kunden, Buchungen und Räder in der folgenden Datenbank:



- a) Sie sollen für folgende Aufgaben die entsprechenden SQL-Anweisungen formulieren.

- aa) Erstellen Sie die Tabelle *Defekt*, welche als Attribut eine *DefektID* und eine *Beschreibung* enthält.

2 Punkte

[illegible]

- ab) Erstellen Sie die Tabelle *DefektBuchung*, welche bis auf das Attribut *Tage* alle Attribute der Tabelle *Buchung* und eine *DefektId* aus der Tabelle *Defekt* enthält.

3 Punkte

[illegible]

- b) Erstellen Sie eine Liste aller Buchungen pro RadTyp für alle Radtypen, zu denen mindestens zehn Buchungen vorliegen.

5 Punkte

RadTypID	Anzahl
1000	23
1001	12
...	

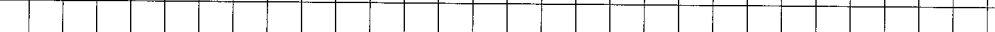
[illegible]

Fortsetzung 5. Handlungsschritt →

Fortsetzung 5. Handlungsschritt

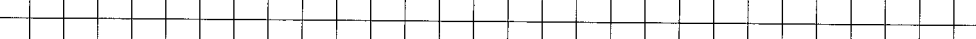
- c) Erstellen Sie eine Liste, in der für jeden Kunden der Gesamtumsatz seiner Buchungen (jeweils Tage * RadTypPreis) aufgelistet ist. Die Liste soll die Datensätze absteigend sortiert nach dem Umsatz enthalten. 5 Punkte

KdID	Umsatz
2002	1400
2001	800
...	...




- d) Geben Sie alle Radtyp-IDs, deren Radtypbezeichnung und Preis an, die einen höheren Preis als der Radtyp ‚Mountainbike‘ haben (RadTypeID = 1001). 5 Punkte

RadTypID	RadTypBez	RadTypPreis
1002	Tandem 500	30
...



- e) Geben Sie für jeden Monat den prozentualen Anteil der Anzahl der Buchungen an der Gesamtanzahl der Buchungen für das Jahr 2019 an. 5 Punkte

Monat	Anteil
1	5
2	7
...	...



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.

9

Belegsatz

Fachinformatiker Anwendungsentwicklung
Fachinformatikerin Anwendungsentwicklung
1196

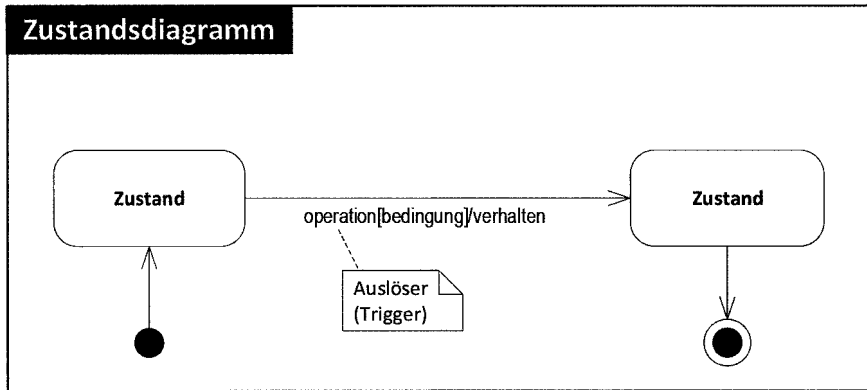
1

Ganzheitliche Aufgabe I Fachqualifikationen

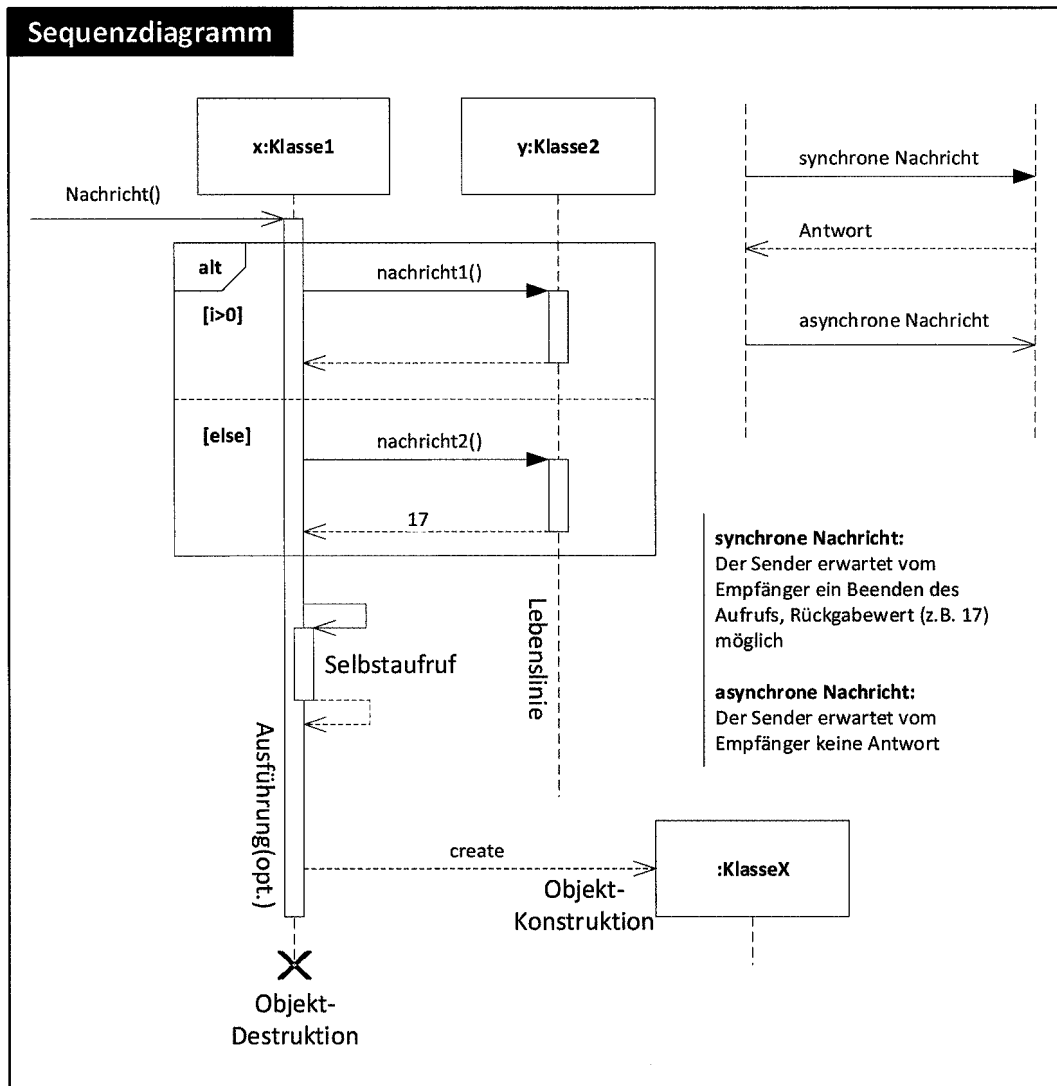
Inhalt

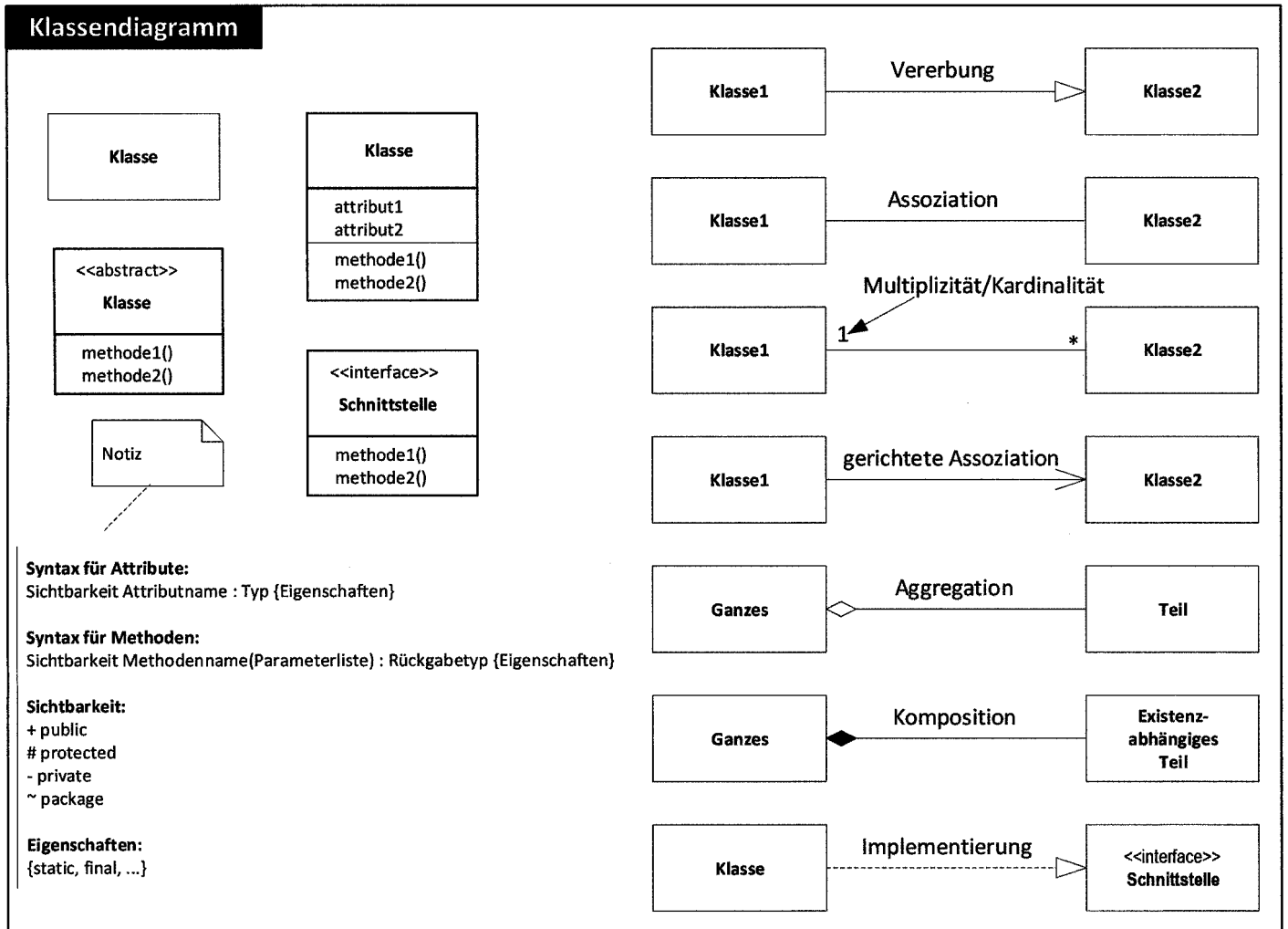
UML-Zustandsdiagramm	Seite 2
UML-Sequenzdiagramm	Seite 2
UML-Klassendiagramm	Seite 3
SQL-Syntax (Auszug)	Seite 4/5

UML-Zustandsdiagramm



UML-Sequenzdiagramm





Syntax	Beschreibung
Tabelle	
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 Datentyp ADD FOREIGN KEY (Spaltenname) REFERENCES Tabellename(Primärschlüsselspaltenname)	Änderungen an 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
Befehle, Klauseln, Attribute	
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.
SELECT DISTINCT	Eliminiert Redundanzen, die in einer Tabellen auftreten können, Werte werden jeweils nur einmal angezeigt.
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
Syntax	Beschreibung
Datenmanipulation	
DELETE FROM Tabellename	Löschen von Datensätzen in der genannten Tabelle
UPDATE Tabellename SET	Aktualisiert Daten in Feldern einer Tabelle
INSERT INTO Tabellename VALUES (Wert für Spalte 1 [, Wert für Spalte 2, ...])	Fügt Datensätze in die genannte Tabelle, die entweder mit festen Werten belegt oder Ergebnis eines SELECT-Befehls sind

SQL-Syntax (Auszug)

oder SELECT ... FROM ... WHERE	
Aggregatfunktionen	
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
Funktionen	
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)
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 2018-03-29