
Prüfungsteilnehmer**Prüfungstermin****Einzelprüfungsnummer**

Kennzahl: _____**Kennwort:** _____**Arbeitsplatz-Nr.:** _____**Frühjahr
2010****46116**

**Erste Staatsprüfung für ein Lehramt an öffentlichen Schulen
— Prüfungsaufgaben —**

Fach: **Informatik (Unterrichtsfach)****Einzelprüfung:** **Softwaretechnologie/Datenbanksysteme****Anzahl der gestellten Themen (Aufgaben):** 2**Anzahl der Druckseiten dieser Vorlage:** 9

Bitte wenden!

Thema Nr. 1

Aufgabe 1: Modellierung eines Reiseunternehmens

Es sei folgender Sachverhalt gegeben:

Ein Reiseunternehmen bietet verschiedene Reisen an. Dazu beschäftigt es eine Reihe von Reiseleitern, wobei eine Reise von mindestens einem Reiseleiter geleitet wird. Da Reiseleiter freiberuflich arbeiten, können sie bei mehreren Reiseunternehmen Reisen leiten.

An einer Reise können mehrere Teilnehmer teilnehmen, ein Teilnehmer kann auch an verschiedenen Reisen teilnehmen.

- a) Modellieren Sie diesen Sachverhalt in einem UML-Klassendiagramm. Für Teilnehmer und Reiseleiter sollen Sie dabei eine abstrakte Oberklasse definieren. Achten Sie dabei auf die Multiplizitäten der Assoziationen. Sie müssen keine Attribute bzw. Methoden angeben.
- b) Eine Reise kann jedoch nur mit einer begrenzten Kapazität angeboten werden, das heißt, zu einer bestimmten Reise kann nur eine begrenzte Anzahl von Teilnehmern assoziiert werden. Als Ausgleich soll pro Reise eine Warteliste verwaltet werden.
Modellieren Sie diesen erweiterten Sachverhalt in einem neuen Diagramm. Nicht veränderte Klassen brauchen nicht noch einmal angegeben werden. Beachten Sie dabei, dass die Reihenfolge bei einer Warteliste eine Rolle spielt.
- c) Implementieren Sie die in Aufgabenteil b) modellierten Klassen in Java. Fügen Sie eine Methode hinzu, die einen Teilnehmer von einer Reise entfernt. Dabei soll automatisch der erste Platz der Warteliste zu einem Reiset Teilnehmer werden, wenn die Warteliste nicht leer ist. Achten Sie auf die Navigierbarkeit Ihrer Assoziationen. Sie können davon ausgehen, dass die Methode nur mit Teilnehmern aufgerufen wird, die in der Tat Teilnehmer der Reise sind.

Fortsetzung nächste Seite!

Aufgabe 2:**1. Relationale Algebra**

Die relationale Algebra wird aufgebaut über einer Grundmenge von mengenwertigen Operationen. Diese Grundoperationen können auf Relationen angewendet werden und erzeugen als Ergebnis wieder eine Relation.

Notation: π = Projektion; σ = Selektion; \bowtie = Join; \times = kartesisches Produkt; \setminus = Mengendifferenz; \cap = Schnittmenge; \cup = Vereinigungsmenge; ρ = Umbenennen; \div = relationale Division

- Beschreiben Sie ein vollständiges SQL-Statement (oder mehrere) und geben Sie an, welche Klausel mit welcher Operation aus der relationalen Algebra korrespondiert (alle Operationen müssen verwendet werden). Bedenken Sie, dass einige Operatoren an mehreren Stellen vorkommen können.
- Definieren Sie die mengenwertige Operation „Division“. Auf welche Grundoperation kann die Division zurückgeführt werden und wie? Ein Beweis ist nicht erforderlich.

2. Transaktionen und Transaktionsverarbeitung

- Nennen und definieren Sie die vier wesentlichen Merkmale einer Datenbanktransaktion.
- Erklären Sie steal, no-steal, force, no-force im Zusammenhang mit der Fehlerbehandlung.
- Fassen Sie die vier Kombinationen von force/no-force, steal/no-steal hinsichtlich der Anforderungen an die Redo- und Undo-Recovery auf folgende Weise zusammen:

	Force	No-Force
Steal	<ul style="list-style-type: none"> • Kein Redo • Undo 	<ul style="list-style-type: none"> • •
No-Steal	<ul style="list-style-type: none"> • • 	<ul style="list-style-type: none"> • •

Übertragen Sie die Tabelle und ergänzen Sie diese.

Fortsetzung nächste Seite!

3. ER-Modellierung und Relationenmodell

Szenario: Sie sollen ein System zur Verwaltung von Firmen entwerfen.

- Eine Firma ist eindeutig bezeichnet durch ihre Handelsregisternummer. Daneben hat sie einen Namen und eine Adresse, die sich aus Straße, Hausnummer, Ort und Postleitzahl zusammensetzt.
- Firmen bestehen aus Abteilungen. Eine Firma kann dabei aus mehreren Abteilungen bestehen, eine Abteilung kann jedoch immer nur zu einer Firma gehören. Es kann Firmen geben, die nicht in Abteilungen eingeteilt sind, aber jede Abteilung gehört zu einer Firma. Abteilungen haben eine eindeutige AbteilungsID und einen Namen.
- Firmen beschäftigen Mitarbeiter. Ein Mitarbeiter kann nur für eine Firma arbeiten, eine Firma aber beliebig viele Mitarbeiter haben. Ein Mitarbeiter hat eine eindeutige Sozialversicherungsnummer (SVN), einen Namen (bestehend aus Name und Vorname), ein Geburtsdatum und ein Alter, das sich aus dem Geburtsdatum errechnet. Daneben hat ein Mitarbeiter eines oder mehrere Spezialgebiete, auf dem bzw. denen er Fachmann ist.
- Mitarbeiter können Angehörige haben. Zu diesen werden jeweils Geburtstag und Name gespeichert. Ein Mitarbeiter kann keine zwei Angehörigen mit gleichem Namen haben.
- Abteilungen bearbeiten Projekte. Ein Projekt ist gekennzeichnet durch seine ProjektID. Daneben besteht es aus einem Namen und einem Kunden. Ein Projekt kann von mehreren Abteilungen bearbeitet werden und eine Abteilung kann an mehreren Projekten arbeiten. Um nicht den Überblick zu verlieren, soll der Zeitraum gespeichert werden, in dem eine Abteilung ein Projekt bearbeitet. Dieser Zeitraum gliedert sich in die Felder „von“ und „bis“.

a) ER-Modellierung:

Erstellen Sie ein ER-Modell des oben beschriebenen Szenarios. Geben Sie Kardinalitäten von Beziehungen sowie Rollennamen an. Verwenden Sie auch abgeleitete, zusammengesetzte und mehrwertige Attribute, mehrstellige Beziehungstypen und schwache Entitätstypen, wo es sinnvoll ist.

b) Relationenmodell:

Überführen Sie das ER-Modell in ein Relationenschema.

Notation: Unterstreichen Sie Primärschlüssel; kennzeichnen Sie Fremdschlüssel durch Angabe der referenzierten Relation in eckigen Klammern.

Beispiel:

Autor(AID, Name)

Buch(ISBN, Schriftsteller[Autor])

Fortsetzung nächste Seite!

4. SQL

Ein Internetauktionenhaus nutzt eine relationale Datenbank zum Speichern seiner relevanten Daten. Angenommen ist folgender Auszug aus dem Tabellenschema zu den Benutzern, Auktionen und Geboten:

User(UId, Name,...)

Auction(AId, Name, Begin_Auction, End_Auction, Seller[User], ...)

Bid(UId, AId, Bid_Time, Amount)

Das Auktionshaus will ein Bewertungssystem einführen, um die Risiken für Verkäufer eindämmen zu können.

Folgende Tabelle soll zu den existierenden Tabellen hinzugefügt werden:

Review(RId, AId[Auction], BuyerID[User], Rating, Rating_Time)

RId, AId und BuyerID sind beliebige ganzzahlige Datentypen, Rating ist ein ganzzahliger Datentyp, der die Werte 1,2,3,4,5 annehmen kann, und Rating_Time ist ein timestamp. RId ist Primärschlüssel und die beiden Fremdschlüssel dürfen nie leer sein.

- a) Geben Sie das vollständige CREATE-Statement zum Anlegen von Review an, das möglichst viele Constraints erfüllt.
- b) Erstellen Sie eine Sicht mit dem Namen Auction_Review mit den Attributen (AId, UId, Durchschnittsrating):
Zu jeder Auktion (gekennzeichnet mit AId) soll das Durchschnittsrating jedes Nutzers (UId) angegeben werden, der auf diese Auktion geboten hat. Die Durchschnittsratings sollen auf allen Ratings des Nutzers basieren, die zeitlich vor Ende der betrachteten Auktion vergeben wurden.
- c) Erstellen Sie eine Sicht mit dem Namen Auction_Max_Amount mit den Attributen (AId, MaxAmount):
Zu jeder abgeschlossenen Auktion (aktueller Zeitpunkt mit sysdate()) soll der größte gebotene Betrag angegeben werden.

Thema Nr. 2

Aufgabe 1: Systementwurf und Programmierung: Objektorientierte Softwareentwicklung

Im Rahmen einer wissenschaftlichen Konferenz können Personen in verschiedenen Rollen auftreten, nämlich als Teilnehmer und/oder als Autor. Eine Konferenz kann von beliebig vielen Autoren und Teilnehmern besucht werden. Autoren reichen eine Veröffentlichung zu einer Konferenz ein. Jede Veröffentlichung wird von entweder zwei oder drei Gutachtern bewertet, die jeweils ein Gutachten über sie verfassen. Ein Gutachten kann positiv oder negativ sein. Jede Veröffentlichung hat mindestens einen Autor. Derselbe Autor kann allerdings mehrere Veröffentlichungen zu einer Konferenz einreichen. Jede Veröffentlichung kann nur bei einer einzigen Konferenz eingereicht werden. Die Gutachter können für mehrere Konferenzen tätig sein. Veröffentlichungen haben einen Titel. Personen haben einen Namen.

- a) Geben Sie ein für diesen Sachverhalt passendes UML-Klassendiagramm an. Verwenden Sie je nach Bedarf uni- und bidirektionale Assoziationen und beschriften Sie die Assoziationsenden mit geeigneten Rollennamen und Multiplizitäten. Die Klassen Ihres Diagramms sollen die im Text genannten Attribute mit jeweils passendem Typ enthalten. Sie können zur Modellierung entweder UML 1.x oder UML 2.0 wählen. Geben Sie an, welchen UML-Standard Sie Ihrer Lösung zugrunde legen.
- b) Implementieren Sie die Klasse `Konferenz` mit einer Methode `findeTeilnehmendeAutoren`, die alle Personen zurückgibt, die sowohl Autor für die Konferenz sind als auch an der Konferenz teilnehmen. Implementieren Sie dazu auch alle Klassen und Assoziationen aus Ihrem Klassendiagramm, die Sie für Ihre Implementierung benötigen.

Der Rückgabetyt der Methode `findeTeilnehmendeAutoren` kann z. B. ein beliebiger Collectiontyp (in Java beispielsweise `Collection`, in C++ beispielsweise `Vector`) sein.

Geben Sie an, welche Sprache Sie gewählt haben.

- c) Gegeben sei folgendes Primärszenario einer erfolgreichen Einreichung zu einer Konferenz:

Ein Autor reicht eine Veröffentlichung bei einer Konferenz ein. Anschließend wird die Veröffentlichung zwei Gutachtern zur Begutachtung zugewiesen, die daraufhin jeweils ein Gutachten erstellen. Beide Gutachten fallen positiv aus. Der Autor wird über die Annahme seiner Veröffentlichung benachrichtigt.

Geben Sie für dieses Szenario ein Sequenzdiagramm an.

Es ist nicht notwendig, dass die verwendeten Methoden im Klassendiagramm aus Teilaufgabe a) enthalten sind.

Fortsetzung nächste Seite!

Aufgabe 2:**1. Integritätsbedingungen**

Eine Aufgabe von Datenbankmanagementsystemen ist die Erhaltung der Datenintegrität.

- a) Welche Arten von Integritätsbedingungen gibt es?
- b) Geben Sie für jede Art von Integritätsbedingung je ein Beispiel an.

2. Modellierung

Der Veranstalter der Ausstellung "Becit" möchte diese in einer relationalen Datenbank verwalten:

Zu jedem Ausstellungsraum gibt es eine eindeutige Raumnummer (RNr). Ein Raum hat eine bestimmte Fläche (in m^2), eine Höhe (in m) und zusätzliche Sonderausstattungen wie Beamer, Fernseher, Stühle, Bänke, usw. Die Ausstellungsstände werden in Räumen aufgestellt. Es hat sich eingebürgert, den Ständen innerhalb von Räumen Buchstaben zuzuordnen (Sid). So ist beispielsweise Stand "3B" der 2. Stand in Raum 3. Zu Ständen ist bekannt, wieviel Fläche (in m^2) sie benötigen, die Anzahl benötigter Tische und Stühle und möglicherweise benötigte Sonderausstattungen des entsprechenden Raumes.

Die Stände werden dann den Anbietern zugeordnet. Für die Anbieter werden eindeutige Ids vergeben. Zusätzlich sind von diesen jeweils Name, Adresse und Telefonnummer bekannt.

Das Personal kümmert sich um den reibungslosen Ablauf der Ausstellung. Jedes Mitglied des Personals besitzt eine Personalnummer (PNr). Ebenfalls bekannt ist der Lohn des Personals.

Das Personal unterteilt sich in Auf-/Abbau-Personal, Sicherheitspersonal und Reinigungspersonal. Das Auf-/Abbau-Personal bekommt bestimmte Stände zugewiesen, die es auf- und abbaut. Das Sicherheitspersonal kümmert sich um die Sicherheit der Ausstellung. Die einzelnen Mitglieder des Sicherheitspersonals patrouillieren hierfür in den ihnen zugewiesenen Ausstellungsräumen. Sie können über ein Funkgerät verfügen und ggf. auch bewaffnet sein. Das Reinigungspersonal kümmert sich um die Endreinigung der Ausstellungsräume. Da hier in verschiedenen Räumen unterschiedlich viel Arbeit anfallen kann, können vorab keine Räume zugeordnet werden.

Zu beachten ist insbesondere, dass sämtliche Personen des Personals auch mehrere Aufgaben übernehmen können, d.h. sie können beim Aufbau mithelfen, zusätzlich während der Ausstellung für die Sicherheit sorgen und am Ende ggf. die Räume mit reinigen.

- a) Entwerfen Sie für das beschriebene Szenario ein ER-Diagramm. Bestimmen Sie hierzu
 - die Entity-Typen, die Relationship-Typen und jeweils deren Attribute;
 - ein passendes ER-Diagramm;
 - die Primärschlüssel der Entity-Typen, welche Sie anschließend in das ER-Diagramm eintragen;
 - die Kardinalitäten der Relationship-Typen, welche Sie ebenfalls in das ER-Diagramm eintragen.

Fortsetzung nächste Seite!

- b) Überführen Sie das ER-Modell aus Aufgabe a) in ein verfeinertes relationales Modell. Geben Sie hierfür die verallgemeinerten Relationenschemata an.
- c) Bei den Zuordnungen der Sonderausstattungen kann es zur Verletzung einer Integritätsbedingung kommen.
- Begründen Sie, woran dies liegt.
 - Wie lässt sich das Problem innerhalb einer Datenbank verhindern?

3. Normalformen

Gegeben sei das folgende Relationenschema

Kaeufe: {[*ArtikelId*, *ArtikelBezeichnung*, *Einkaufspreis*, *Verkaufspreis*, *Datum*,
TagesAktion, *KundeName*, *KundeVorname*, *KundeGebdatum*, *Filiale*, *Verkaeuer*]}

und eine Menge F von funktionalen Abhängigkeiten. **Hinweis:** Die Abkürzungen innerhalb der Menge F beziehen sich auf die Großbuchstaben in den Attributnamen der Relation *Kaeufe*.

$$F := \{ \begin{array}{l} AI \rightarrow AB \\ AI, AB \rightarrow EP \\ KN, KV \rightarrow KG \\ AI, D, KN, KV \rightarrow V \\ AI, AB, KN, KV, D, V \rightarrow F, VP \\ D \rightarrow TA \\ V \rightarrow F \end{array} \}$$

- Ermitteln Sie sämtliche Schlüsselkandidaten von *Kaeufe*.
- Zeigen Sie, dass das Relationenschema *Kaeufe* nicht in dritter Normalform ist.
- Bestimmen Sie eine kanonische Überdeckung F_c zu F .

4. SQL und Relationale Algebra

Gegeben sei folgendes Schema der relationalen Datenbank eines Aktienhändlers:

Aktie{*ANr*, *Name*, *Firma*[*Firma*], *Ausgabedatum*, *Ausgabepreis*}

Aktienkurs{*ANr*[*Aktie*], *Datum*, *Zeit*, *Wert*}

Firma{*Name*, *Adresse*, *Telefon*}

Kunde{*KndNr*, *Name*, *Geburtsdatum*, *Telefon*}

Konto{*KntNr*, *KndNr*[*Kunde*], *Eroeffnungsdatum*, *Saldierungsdatum*}

Fortsetzung nächste Seite!

$Buchung\{KntNr[Konto], BNr, ANr[Aktie], Aktienzahl, EKdatum, EKzeit, EKpreis, VKdatum, VKzeit, VKpreis\}$

Hinweis: Für Fremdschlüssel wird nach dem Format $\langle Attr \rangle [\langle RefTable \rangle]$ die referenzierte Tabelle angegeben. Beispielsweise ist $KndNr[Kunde]$ in der Relation *Konto* ein Fremdschlüssel der Relation *Kunde*.

Sämtliche verfügbaren Aktien sind in der Relation *Aktie* enthalten. Die zugehörigen Kurse werden in der Relation *Aktienkurs* festgehalten. Zu jedem (im Rahmen dieser Aufgabe notwendigen) Zeitpunkt existiert ein korrespondierendes Tupel. Die Aktiengesellschaften (also Firmen, welche Aktien ausgeben) sind in der Relation *Firma* gespeichert. Die Kunden des Aktienhändlers werden in der Relation *Kunde* verwaltet. Jeder Kunde kann beliebig viele Konti eröffnen. Diese werden in der Relation *Konto* aufgeführt und dem Kunden direkt zugeordnet. Aktienkäufe und Aktienverkäufe erfolgen paketweise. Kauft ein Kunde ein Paket einer Aktie, so wird ein neues Tupel in der Relation *Buchung* erstellt. Die Buchungsnummern werden für jedes Konto jeweils in aufsteigender Reihenfolge beginnend bei 1 vergeben. Innerhalb eines neuen Tupels werden Umfang des Pakets (Aktienzahl), Datum und Zeitpunkt des Kaufs und insbesondere der Preis des Aktienpakets festgehalten. Die entsprechenden Attribute für einen Verkauf werden zu *NULL* initialisiert. Wird später ein Verkauf des Aktienpakets durchgeführt, so werden die Werte für *VKdatum*, *VKzeit* und *VKpreis* entsprechend gesetzt. Aktuell sei der Zeitpunkt 13:00:00 Uhr und das Datum 2008/07/20. Für Datumsangaben gelte insbesondere, dass eine Verschiebung des Datums um t Tage durch folgende Addition durchgeführt werden kann: $2008/07/20 + t$. Beispielsweise ergibt $2008/07/20 + 20 = 2008/08/09$.

- a) Formulieren Sie die Anfragen in den folgenden Teilaufgaben (i - ii) in der Anfragesprache SQL. Zur Vereinfachung dürfen Sie sich beliebige Sichten (Views) definieren.
- Gesucht sind alle Aktienanteile, welche Kunde "Müller" aktuell hält. Geben Sie für diese Kontonummer, Buchungsnummer, Aktiennummer und die zugehörigen Firmennamen aus.
 - Nehmen Sie an, das Ergebnis aus Teilaufgabe i) sei als View *AktuelleAktienMueller* verfügbar.
Geben Sie nun gruppiert nach den Firmennamen die Gewinne/Verluste aus, welche Müller beim Verkauf seiner aktuellen Aktienanteile machen würde.
- b) Formulieren Sie folgende Anfrage in der relationalen Algebra:
- Für welche Firmen hat sich der Aktienkurs im 1. Jahr nach der Ausgabe mindestens verdoppelt?